Chapter 10

Affine Functions: Global Analysis

While, in Local Analysis, we dealt with local problems in that, given an input, we asked questions about features of the output for nearby inputs, in Global Analysis, we will deal with global problems in that we will not be given any particular input but, on the contrary, one way or the other, we will have to consider all inputs. More precisely:

- We will establish theorems about the way affine functions deal with any and all inputs.
- We will discuss reverse problems, that is, problems in which, given an affine function, and given a local feature that the outputs should have, we will find the input(s), if any, whose outputs have that given feature.
- We will get the global input-output rule of a function that meets given requirements. In other words, this is a problem of finding a function specified indirectly as a solution of a functional equation subject to some additional conditions. Depending on the nature of these additional conditions then, there are two main kinds of such problems:
  - Initial Value Problems in which the input is thought of as standing
for time and where the additional conditions involve the outputs of the Taylor functions at some initial time.

**Example 1.** Consider the problem of finding the trajectory of a bullet. This will be a solution of Newton’s equation which deals with the effect of gravity. But we also need additional conditions: We need to know where the bullet was at the initial time, that is when the gun was fired, and we also need to know how fast the bullet is going and in what direction when it leaves the gun. After that, Newton’s equation will take care of where the bullet will be at any time inasmuch as it will specify the global input-output rule that, for any input time, returns the output position of the bullet.

Boundary Value Problems in which the input is thought of as standing for space and where the additional conditions involve the outputs of the Taylor functions at both some departure point and some arrival point.

**Example 2.** Consider the problem of getting a mountain road across a river canyon. The shape of the bridge will be the result of solving complicated equations. But the additional conditions are that the bridge to be built will have to match at both ends the height and the slope of the road.

Perhaps surprisingly, local issues and global issues have very little in common. In fact, local information usually does not translate into global information and, vice versa, global information does not translate into local information.

**Example 3.** Knowing the average age of the students in a school does not say anything about the average age of the students in a given classroom. Knowing the average age of the students in a classroom obviously does not say anything about the average age of the students in the school but what is perhaps more surprising, knowing the average age of the students in each and every classroom still does not give us the average age of the students in the school.

As we will now see, affine functions are truly exceptional in that we will be able to solve all global problems exactly. With most other functions, though, we will be able to solve only some global problems and mostly only approximately so.

### 10.1 The Essential Question

As when we set out to investigate any kind of functions, the first thing we must do is to find out if the offscreen graph of a cubic function consists of just the local graph near $\infty$ or if it also includes the local graph near one or more $\infty$-height inputs.

**Example 4.** When we set out to describe a planet, say in a science fiction, we certainly must describe its sun.
10.2. CONSTANT APPROXIMATIONS

In other words, we need to ask the **Essential Question:**

- Do all *bounded inputs* have *bounded outputs* (for some extent of the output ruler)

or

- Is there one (or more) bounded input that is an $\infty$-height input, namely whose nearby inputs have infinite outputs (no matter what the extent of the output ruler)?

In the case of *affine functions,* the answer to the **Forced Question** is:

**THEOREM 1** (*Bounded Height*). For affine functions, all bounded inputs return bounded outputs.

The immediate consequence of the **Bounded Height Theorem** is:

**THEOREM 2** (*Offscreen Graph*). For any affine function, the offscreen graph consists of just the local graph near $\infty$.

**APPROXIMATION THEOREMS**

These theorems will in fact describe the *smoothness* of (the graph of) an affine function $QUADRATIC_{a,b,c}$ by way of **approximating functions.**

They are *global theorems* in that they describe, not just the way the local graph looks near a *given* input, but the way the local graph looks near *any* input.

**10.2 Constant Approximations**

We saw in the previous chapter that the *local input-output rule* of $AFFINE_{a,b}$ near a given input $x_0$ is

\[
h \xrightarrow{AFFINE(x_0)} AFFINE(x_0 + h) = \left[ax_0 + b\right] + \left[a\right]h
\]

and that the successive terms in the local input-output rule are in order of diminishing size.
CHAPTER 10. AFFINE FUNCTIONS: GLOBAL ANALYSIS

1. Since what determines the height near $x_0$ is the local constant coefficient $[ax_0 + b]$, and since the local constant coefficient depends on $x_0$, the height of the local graph will be different everywhere, that is for all inputs. More precisely, since the local constant term, $[ax_0 + b]$ is the most size-important term, we have:

**THEOREM 3 (Local Constant Approximation).** Affine functions are locally approximately constant. More precisely

$$x_0 + h \xrightarrow{\text{AFFINE}} \text{AFFINE}(x_0 + h) = \text{AFFINE}(x_0) + [...]$$

with the immediate consequence that affine functions can be locally approximated by constant functions. However, since the local constant coefficient of $\text{AFFINE}_{a,b}$ near $x_0$ depends on $x_0$, affine functions cannot be globally approximated by a single constant function.

Indeed, the fact that the output remains approximately constant near $x_1$ and that the output remains approximately constant near $x_2$ does not mean that the output for inputs near $x_1$ and the output for inputs near $x_2$ will be the same and of course will not.

**EXAMPLE 5.** When we look at a clock for, say, a half a minute, it seems as if the hands are not moving and that the time showed by the clock remains the same. And while that is true regardless of when we are looking at the clock, we cannot say that the time showed by the clock remains always the same. That is true only “locally”.

2. The **Local Constant Approximation Theorem** does not say how different the heights of the local graphs will be but the following mitigates matters:

**THEOREM 4 (Continuity).** Under affine functions, inputs that are near each other give outputs whose heights are near each other.

To make the case for the **Continuity Theorem**, the idea that comes immediately to the mind is, given two inputs $x_1$ and $x_2$ that differ by $h$, say $x_2 - x_1 = h$, just to compute the difference $\text{AFFINE}(x_2) - \text{AFFINE}(x_1)$:

$$\text{AFFINE}(x_2) - \text{AFFINE}(x_1)$$

$$= [ax_2 + b] - [ax_1 + b]$$

$$= [ax_2 - ax_1] + [b - b]$$

$$= a [x_2 - x_1]$$
and since \( x_2 - x_1 = h \)

\[ = ah \]

and since \( \text{bounded} \cdot \text{small} = \text{small} \)

\[ = [...] \]

So, the difference between the \textit{heights} will be \textit{small}.

Here is a different way to make the case for the \textbf{Continuity Theorem}:

Since \( x_2 - x_1 = h \), we have that \( x_2 = x_1 + h \) and

\[ x_2 = x_1 + h \rightarrow \text{AFFINE}(x_2) = \text{AFFINE}(x_1 + h) \]

and therefore, by the \textbf{Local Constant Approximation Theorem},

\[ \text{AFFINE}(x_2) = \text{AFFINE}(x_1) + [...] \]

so that

\[ \text{AFFINE}(x_2) - \text{AFFINE}(x_1) = [...] \]

The difference between the two proofs is that the second one does not involve the local coefficients.

\textbf{Example 6.} Given the affine function \( \text{MARA}_{+3,+17} \), that is given the function whose global input-output rule is

\[ x \xrightarrow{\text{MARA}} \text{MARA}(x) = +3x + 17 \]

and given the input, \(-5\), we localize the function \( \text{MARA} \) at \((-5)\) as follows:

\[ x|_{x=-5+h} \xrightarrow{\text{MARA}} \text{MARA}(x)|_{x=-5+h} = +3x + 17|_{x=-5+h} \]

\[ = 3(-5 + h) + 17 \]

\[ = -15 + 3h + 17 \]

and, collecting terms of the same order of magnitude,

\[ = +2 + 3h \]

We can therefore say that as long as \( x \) remains near \(-5\), the output remains near \(+2\). In other words, as long as \( x \) remains near \(-5\), the output remains approximately constant.

\textbf{3.} In turn, as a consequence of the \textbf{Continuity Theorem}, we have:
CHAPTER 10. AFFINE FUNCTIONS: GLOBAL ANALYSIS

THEOREM 5 (No Jump). Affine functions have no jumps.

The case for the No Jump Theorem is also easy to make: Say you wanted to know if there is a jump at $x_0$. (See Chapter 3 for what we mean by “jump”.) Take two inputs near $x_0$, one on each side of $x_0$. But then, if there really were a jump at $x_0$, then the outputs for the two inputs would be separated by approximately the size of the jump which, since the size of a jump is bounded, would prevent the outputs from being near each other which they have to by the Continuity Theorem since both inputs are near $x_0$.

4. Finally, affine functions can be globally approximated by piecewise constant functions, something which, later on, will turn out to be extremely important.

EXAMPLE 7. The affine function whose bounded graph is given on the left is globally approximated by the piecewise constant function whose graph is given on the right:

inasmuch as the difference between the output of the cubic function and the output of the piecewise constant function is always less than 0.5. Observe, though, that the approximating function has jumps. No such thing as a free lunch.

EXAMPLE 8. When we look at a clock for, say, a half a minute, it seems as if the hands are not moving and so that the time that the clock shows remains the same. And while that is true regardless of when we are looking at the clock, we cannot say that the time that the clock shows remains the same. That is true only “locally”.

10.3 Affine Approximations

We saw in the previous chapter that the local input-output rule of $\text{AFFINE}_{a,b}$ near a given input $x_0$ is
10.3. AFFINE APPROXIMATIONS

global slope

\[
h \xrightarrow{AFFINE(x_0)} AFFINE(x_0 + h) + [ax_0 + b] + [a]h
\]

and that the successive terms in the local input-output rule are in order of diminishing size.

1. Since what determines the slope near \( x_0 \) is the local linear coefficient \( [a] \), and since the local linear coefficient does not depend on \( x_0 \), we have

**THEOREM 6 (Local Affine Approximation).** Affine functions are locally exactly affine. More precisely

\[
x_0 + h \xrightarrow{AFFINE} AFFINE(x_0 + h) = AFFINE(x_0) + AFFINE'(x_0)h
\]

Since the slope is the same everywhere, we will say that the affine function \( AFFINE_{a,b} \) has a **global slope** given by the linear coefficient \( a \).

While it is hardly surprising that affine functions can be locally approximated by affine functions, and while the name of the theorem is thus somewhat misleading, the theorem is stated here to be contrasted with the corresponding theorems in the case of quadratic functions and cubic functions.

2. Even though the **Local Affine Approximation Theorem** does say how different the slopes of the local graphs will be (they will be the same), the following will be interesting in its own right and in comparison with the other polynomial functions:

**THEOREM 7 (Differentiability).** Under affine functions, inputs that are near each other give outputs where slopes are near each other. (They are of course the same.)

We now look at how to make the case for the **Differentiability Theorem.**

- The idea that comes to mind immediately is, given two inputs \( x_1 \) and \( x_2 \) that differ by \( h \), say \( x_2 - x_1 = h \), just to compute the difference \( AFFINE'(x_2) - AFFINE'(x_1) \). The computations go exactly as in the case of the **Continuity Theorem** except quite a bit simpler:

\[
AFFINE'(x_2) - AFFINE'(x_1) = [a - a] = 0
\]

So, we have shown that the difference between the slopes not only will be small but that it is in fact 0.
Here is a different way to make the case for the Differentiability Theorem that does not involve directly the local coefficients.

- Since \( x_2 - x_1 = h \), we have that \( x_2 = x_1 + h \) and

\[
x_2 = x_1 + h 
\xrightarrow{AFFINE} \quad AFFINE(x_2) = AFFINE(x_1 + h)
\]

and therefore, by the Local Affine Approximation Theorem, that

\[
AFFINE(x_2) = AFFINE(x_1) + AFFINE'(x_1)h
\]

- Since \( x_2 - x_1 = h \), we have that \( x_1 = x_2 - h \) and

\[
x_1 = x_2 - h \xrightarrow{AFFINE} \quad AFFINE(x_1) = AFFINE(x_2 - h)
\]

and therefore, by the Local Affine Approximation Theorem, that

\[
AFFINE(x_1) = AFFINE(x_2) - AFFINE'(x_2)h
\]

Adding side by side, we get

\[
\begin{align*}
AFFINE(x_2) + AFFINE(x_1) & = AFFINE(x_1) + AFFINE(x_2) + AFFINE'(x_1)h - AFFINE'(x_2)h \\
\end{align*}
\]

that is

\[
0 = [AFFINE'(x_1) - AFFINE'(x_2)]h
\]

so that, since, as small as it is, \( h \neq 0 \), we must have

\[
[AFFINE'(x_2) - AFFINE'(x_1)] = 0
\]

**NOTE.** The difference between the two proofs is very much like, in programming, the difference between “machine language” and “high level languages” in which the nitty-gritty details are hidden. Here, it is the concept of derivative that hides the local coefficients.

3. In turn, as a consequence of the Differentiability Theorem, we have:

**THEOREM 8 (No Kink).** Affine functions have no kinks.
Here, the case is extremely to make: if a cubic function had a kink at \( x_0 \), then it could not be approximated by a single affine function. (Each side of the graph near \( x_0 \) could, but by different affine functions.)

**FORCED FEATURE-SIGN CHANGES**

The forced feature-sign changes are the feature-sign changes that are forced by the offscreen graph which, by the **Bounded Height Theorem** for cubic functions consists of only the local graph near \( \infty \). So, what forces the forced feature-sign changes are the feature-signs near \( \infty \) as given by the **Local Features Near \( \infty \) Theorem** in Chapter 13.

More precisely, given an affine function \( AFFINEa, b \), that is the cubic function specified by the global input-output rule

\[
\begin{align*}
  x & \xrightarrow{AFFINE} Affine(x) = ax + b
\end{align*}
\]

we will look separately at the two features.

### 10.4 Forc€d Slope-sign Change

When \( x \) is near \( \infty \) the **Local Features Near \( \infty \) Theorem** says that (Keep in mind the Magellan view.):

- When \( a \) is \( + \), \( x \) near \( \infty \) \( \text{Slope-Sign} = (\backslash, \backslash) \)
- When \( a \) is \( - \), \( x \) near \( \infty \) \( \text{Slope-Sign} = (\backslash, \backslash) \)

Since, whatever the sign of \( a \), the slope does not change sign at \( \infty \), there is no need for a bounded input for the slope to change sign back, and therefore the offscreen graph does not force the existence of a bounded input \( x \) Slope-sign change.

**Example 9.** Given an affine function whose offscreen graph is
there is no need for a bounded slope-sign change input, \( x^{\text{Slope-sign change}} \), and therefore we can have

\[
\begin{array}{ccc}
\text{Output Ruler} & \text{Input Ruler} \\
+\infty & -\infty \\
-\infty & +\infty \\
\end{array}
\]

And so:

**THEOREM 9 (Forced Slope-Sign Change Non-Existence).** An affine function has no Forced Slope-Sign Change input.

One must realize, though, that the **Forced Slope-Sign Change Non-Existence Theorem** does not say that affine functions cannot have a Slope-sign Change input. As far as we know at this point, an affine function may have a Slope-sign Change input and we will deal with this issue later on.

We also have

**THEOREM 10 (Forced 0-Slope Non-Existence).** An affine function has no Forced 0-Slope input.

One must realize, though, that the **Forced 0-Slope Non-Existence Theorem** does not say that affine functions cannot have a 0-Slope input. As far as we know at this point, an affine function may have a 0-Slope input and we will deal with this issue later on.

### 10.5 Forced Extremum

From the optimization viewpoint, the most immediately striking feature of an affine function is the **non-existence** of a forced extremum input, that is of a bounded input whose output would be either larger than the output of nearby inputs or smaller than the output of nearby inputs.

**THEOREM 11 (Forced Extremum Non-Existence).** An affine function has no forced extremum input.
10.6. **FORCED HEIGHT-SIGN CHANGE**

Again, the **Forced Extremum Non-Existence Theorem** does not say that an affine function cannot have an extremum input, just that the offscreen graph does not force an extremum input. As far as we know at this point, an affine function may have an extremum input and we will deal with this issue later on.

### 10.6 Forced Height-sign Change

When \( x \) is near \( \infty \) the **Local Height Near \( \infty \) Theorem** says that (Keep in mind the Magellan view.):

- When \( a \) is \( + \), Height-Sign | \( x \) near \( \infty \) = \((+,-)\)
- When \( a \) is \( - \), Height-Sign | \( x \) near \( \infty \) = \((-,+))\)

But then, since the height changes sign when \( x \) goes from \(+\infty\) to \(-\infty\) across \( \infty \), there also has to be a bounded input \( x_{\text{Height-sign change}} \) as \( x \) goes from the left side to the right side of the input ruler for the height to change sign back.

In other words, the offscreen graph forces the existence of a bounded input \( x_{\text{Height-sign change}} \). We will say that \( x_{\text{Height-sign change}} \) is a **forced Height-sign change** input.

Moreover, because there is no bounded infinite-height input where the height could change sign, \( x_{\text{Height-sign change}} \) has to be a bounded input where the height is 0.\(^1\) As a result, we have that

\[
x_{\text{Height-sign change}} = x_{0\text{-height}}
\]

**Example 10.** Given the affine function whose offscreen graph is

\(^1\)Here, Educologists are sure to point out that we should also eliminate the possibility of other inputs such as *kinks*, *cusps*, etc to which we can only reply that, in the present context, such inputs are a lot less than even unlikely.
there has to be a bounded height-sign change input:

Of course, just because we found that any affine function has to have a Height-sign change input does not mean that we have a way of locating height-sign change.

10.7 Forced Bounded Graph By Interpolation

As a consequence of the Bounded Height Theorem for affine functions, the offscreen graph consists only of the local graph near \( \infty \) and we can obtain the forced bounded graph by interpolating smoothly the local graph near \(-\infty\) to the local graph near \(+\infty\).

There remains however a question namely whether the bounded graph is straight that is has no concavity.

Since the local graph near \(+\infty\) and the local graph near \(-\infty\) are straight, (i.e. have no concavity), it would seem that it ought to follow necessarily that the forced bounded graph is also straight, that is has no concavity either.

But for this to be the case, we need to make sure that the local graph near \(+\infty\) and the local graph near \(-\infty\) are “lined up”. Otherwise, we would need either join with a straight line but with kinks at the transitions or the joining graph would have to have concavity.

**Example 11.** Given the local graph

\[ \text{Output Ruler} \]
\[ \text{Input Ruler} \]
\[ \text{Screen} \]
\[ \text{Offscreen Space} \]

\[ \text{Mercator view} \]
\[ \text{Magellan view} \]
the joining graph must be

But since the local graph near $\infty$ is that of an exceptional power function
the two pieces do line up and so the forced bounded graph need not kink or
be concave and the forced qualitative global graph can therefore be a straight
line.

10.8 Forced Bounded Graph By Extrapolation

In the case of affine functions, it happens that we can also obtain the forced
bounded graph in a way that is completely different from the way in which
we interpolated the local graph near $\infty$ in that it is on the basis of the Local
Height Theorem and the Local Slope Theorem, :

We start from the local graphs near a number of finite points as follows:

1. We construct local graphs near, say, three different finite inputs, $x_1,
x_2, x_3$. On the basis of the above two theorems, they would look something
like this:
2. However, this is not possible because that would mean that inputs such as \( x_4 \) would have two outputs:

whereas the input-output rule will pair the input \( x_4 \) with only one output, namely \( ax_4 + b \).

3. As a result, the local graphs near finite inputs must all line up and so the bounded graph must be a straight line:

Of course, the bounded graph must line up with the local graph near \( \infty \) as, otherwise, there would have to be a jump in the transition zone.

As we will see, though, in the case of more complicated functions, getting an forced bounded graph by extrapolating local graphs near finite inputs will usually not work.
10.9. EQUATION PROBLEMS

The reader should probably only skim these sections just so as to have an idea later on where to look and then study them as needed.

REVISITING EQUATIONS

In order to deal with location problems, it is necessary to have a tool that works no matter what. The purpose here, then is:

- To make sure of the algebra that we will need to deal with location problems,
- To have a systematic procedure to refer to,
- To set a language that is consistent

10.9 Equation Problems

1. We begin with the terminology that we will use to be extremely clear as to what we are doing. An equation problem will consist of:
   i. A data set from which the number are to be picked/
   EXAMPLE 12. If the problem to be dealt with involves the age of human beings, the data set should probably not involve negative numbers and probably not go much farther than about 100.
   ii. An equation that will specify the solution subset of the data set, that is which will select the numbers in the data set that are to be retained as solutions, that is the numbers that turn the equation into a true sentence. Those numbers in the data set that turn the equation into a false sentence will be called non-solutions.

2. The approach that we will follow, which we will call the REDUCTION METHOD, will be to reduce the original equation to an equation until we get to an equation that we already know how to solve and we will call that equation the reduced equation. Of course, the reduced equation will have to be equivalent to the original equation in the sense that the reduced equation will have to have the same solution subset as the original equation. What we will do here will only be to add the same number to both sides or multiply both sides by the same number (other than 0) so that the following will apply:

THEOREM 12 (Fairness). Given any equation, as long as, whatever we do onto one side of the verb =, we do exactly the same onto the other side of the verb =, we get an equivalent equation.

NOTE. While the Fairness Theorem seems obviously true, making the case that it is true is not that easy because what is not obvious is on what evidence to base the case.
We will thus leave this issue for when the reader takes a course in Mathematical Logic.

3. In the case of affine equations, that is of equations of the form

\[ ax + b = c \]

the Reduction Method proceed as follows:

\[
\begin{align*}
ax + b &= c \\
ax + b - b &= c - b \\
ax &= c - b \\
\frac{ax}{a} &= \frac{c - b}{a} \\
x &= \frac{c - b}{a}
\end{align*}
\]

so that the Fairness Theorem applies and since \( \frac{c - b}{a} \) is the solution of the reduced equation it is also the solution of the original equation.

### 10.10 Inequation Problems

This involves a general procedure that we will call the Pasch Procedure (after the name of the mathematician who first noticed that, while it was quite obvious that in order to go from one side of a point on a straight line to the other side you had to “get across” the point, this turned out to be impossible to “prove” because the question was “on the basis of what” so that it had to be accepted as an axiom).

**Example 13.** Given the point \( P \) on one side of the point \( B \) and the point \( Q \) on the other side of the point \( B \), to go from \( P \) to \( Q \), we need to get across the point \( B \)

1. Roughly, given an inequation problem, that is a data set and an inequation, in order to determine the solution subset of that inequation problem, we will proceed in two stages:

I. We will locate the boundary of the solution subset of the inequation problem by finding the solution subset of the associated equation problem.

(In the case of an affine inequation, we have already seen in the previous section how to deal with this stage.)

To graph the boundary point we will use:
10.10. INEQUATION PROBLEMS

• a solid dot when the boundary point is a solution of the inequation
• a hollow dot when the boundary point is a non-solution of the inequation.

II. We will locate the interior of the solution subset of the inequation problem, that is the solution subset of the associated strict inequation problem.

In the case of an affine inequation, the boundary consists of only one boundary point which separates the data set in two sections which we will call Section A and Section B. Then, we will locate the interior as follows:

i. We will pick a test number in Section A and check if the test number is a solution or a non-solution of the given inequation.

ii. We will pick a test number in Section B and check if the test number is a solution or a non-solution of the given inequation.

iii. We will then conclude with the help of

THEOREM 13 (Pasch).

• If the test number in a section is a solution, then all numbers in that same section are included in the solution subset.
• If the test number in a section is a non-solution, then all numbers in that same section are non-included in the solution subset.

EXAMPLE 14. Given the inequation problem in which
• the data set consists of all numbers
• the inequation is

\[ x \geq -13.72 \]

we locate separately.

i. The boundary point of the solution subset of the inequation problem is the solution of the associated equation:

\[ x = -13.72 \]

which, of course, is \(-13.72\) and which we graph as follows since the boundary point is a solution of the inequation.

![Boundary point](image)

ii. The interior of the solution subset, that is the solution subset of the associated strict inequation

\[ x > -13.72 \]

i. The boundary point \(-13.72\) separates the data set in two sections, Section A and Section B:
ii. We then test each section:

- We pick $-1000$ as test number for Section A because, almost without a glance we know $-1000$ is going to be in Section A and because it is easy to check in the inequation: we find that $-1000$ is a non-solution so that, by Pasch Theorem, all numbers in Section A are non-solutions.

- We pick $+1000$ as test number for Section B because, almost without a glance we know $+1000$ is going to be in Section B and because it is easy to check in the inequation: we find that $+1000$ is a solution so that, by Pasch Theorem, all numbers in Section A are solutions.

While, in the case of affine inequations, we can work directly on the inequation, this is not generally the case while, as we will see, the advantage of the Pasch Procedure is that it will work in all cases.

**Location Theorems**

### 10.11 Locating Inputs Whose Output $= y_0$

The simplest global problem is when the feature that is required from the output is to be some given number $y_0$. In other words, given a function, we ask for the input(s), if any, whose output is $y_0$. (A particular case of this problem is to find the 0-height input(s), if any.)

What we do is:

i. We write the global input-output rule

$$x \xrightarrow{A_{a,b}} A_{a,b}(x) = ax + b$$
ii. Since we want the output to be equal to $y_0$, that is since we want

$$A_{a,b}(x) = y_0$$

we set the equation

$$ax + b = y_0$$

iii. We then solve using the Reduction Method:

\[
\begin{align*}
ax + b &= y_0 \\
ax + b - b &= y_0 - b \\
ax &= y_0 - b \\
ax &= \frac{y_0 - b}{a} \\
x &= -\frac{y_0 - b}{a}
\end{align*}
\]

So, $x_{y_0 \text{-height}} = \frac{y_0 - b}{a}$

**Example 15.** Given the affine function whose global input-output rule is:

$$x \xrightarrow{A_{a,b}} A_{a,b}(x) = -2.45x + 43.72$$

find $x_{0 \text{-height}}$

**Note.** The resulting fraction, though, is only “code” that indicates what division is to be made and one should keep in mind that most division do not end by themselves so that the result of a division is usually only approximate.

We set the output equal to 0 and solve the resulting equation:

\[
\begin{align*}
-2.43x + 43.72 &= 0 \\
-2.43x &= 43.72 - 43.72 \\
-2.43x &= 0 - 43.72 \\
-2.43x &= -43.72 \\
\frac{-2.43x}{-2.43} &= \frac{-43.72}{-2.43} \\
x &= +\frac{43.72}{2.43}
\end{align*}
\]

So, $x_{0 \text{-height}} = +\frac{43.72}{2.43}$

If we want $x_{0 \text{-height}}$ with two decimals, the division gives 17.99 and we can write either:

- $x_{0 \text{-height}} = 17.99 + [...]$
  - where [...] stands for something positive too small to matter here,
or

\[ x_0 \text{-height} = 18.00 + [...] \]

where [...] stands for something negative too small to matter here.

If we want \( x_0 \text{-height} \) with five decimals, the division gives 17.99176 and we can write either:

\[ x_0 \text{-height} = 17.99176 + [...] \]

where [...] stands for something positive too small to matter here,

or

\[ x_0 \text{-height} = 17.99177 + [...] \]

where [...] stands for something negative too small to matter here.

10.12 Locating Inputs Whose Output > \( y_0 \) Or < \( y_0 \)

Given the affine function \( A_{a,b} \), we are now ready to deal with the global problem of finding all inputs whose output is smaller (or larger) than some given number \( y_0 \).

The input-output rule of \( A_{a,b} \) is:

\[ x \xrightarrow{A_{a,b}} A_{a,b}(x) = ax + b \]

If we want, for instance, the output to be larger than \( y_0 \), we write

\[ A_{a,b}(x) > y_0 \]

which means that we have to find the solution subset of the inequation

\[ ax + b > y_0 \]

So, according to the Pasch Procedure,

I. We find the boundary point of the inequation problem as the solution of the associated equation:

\[ ax + b = y_0 \]

So, we solve using the Reduction Method:

\[
\begin{align*}
ax + b &= y_0 \\
ax + b - b &= y_0 - b \\
ax &= y_0 - b \\
\frac{ax}{a} &= \frac{y_0 - b}{a}
\end{align*}
\]
10.12. LOCATING INPUTS WHOSE OUTPUT $> Y_0 \ OR < Y_0$

\[ x = \frac{y_0 - b}{a} \]

which we graph with a hollow dot since, here, the inequation is \textbf{strict} (i.e. $<\) so that the boundary point $\frac{y_0 - b}{a}$ is a non-solution of the inequation:

\[ \text{Boundary point} \]

\[ -\infty \quad \frac{y_0 - b}{a} \quad +\infty \]

II. We determine the \textit{interior} of the solution subset of the inequation problem by \textit{testing}.

Since the boundary point $\frac{y_0 - b}{a}$ separates two sections in the \textit{data set} (i.e. all inputs), Section A and Section B,

\[ \text{Section A} \quad \frac{y_0 - b}{a} \quad \text{Section B} \]

we must test each section but since, as we shall see, all that matters will be the sign of $a$, it will be easier to deal separately with the case when $a$ is \textit{positive} and the case when $a$ is \textit{negative}:

- When $a$ is \textit{positive},
  - We pick $-\infty$ to test Section A because we know $-\infty$ is in Section A and because it is easy to check $-\infty$ in the inequation:
    \[ ax + b|_{x=-\infty} > y_0 \]
    \[ a(-\infty) + [...] > y_0 \]
    and since $a$ is positive,
    \[ -\infty + [...] > y_0 \]

Since this is \textbf{false}, $-\infty$ is a non-solution and, by \textbf{Pasch Theorem}, all inputs in Section A are also non-solutions:
We pick $+\infty$ to test Section B because we know $+\infty$ is in Section B and because it is easy to check $+\infty$ in the inequation:

\[
ax + b |_{x = +\infty} > y_0 \\
 a(+\infty) + [\ldots] > y_0
\]

and since $a$ is positive,

\[
+\infty + [\ldots] > y_0
\]

Since this is true, $+\infty$ is a solution and, by Pasch Theorem, all inputs in Section B are also solutions:

\[\begin{array}{c}
\text{Non-solutions} \\
-\infty \rightarrow \text{Section A} \\
\uparrow \\
\text{Solutions} \\
\downarrow \\
\uparrow \\
\text{Section B} \\
\rightarrow \text{Non-solutions} \\
+\infty
\end{array}\]

- When $a$ is negative, proceeding in the same manner as above we find that the solution subset of the inequation problem is

\[\begin{array}{c}
\text{Solutions} \\
-\infty \rightarrow \text{Section A} \\
\uparrow \\
\text{Non-solutions} \\
\downarrow \\
\uparrow \\
\text{Section B} \\
\rightarrow \text{Solutions} \\
+\infty
\end{array}\]

### 10.13 Initial Value Problem

An **Initial Value Problem** asks the question:

- What is the input-output rule of a function $F$ given that:
  - The function $F$ is affine
  - The slope of the function $F$ is to be a given number $a$
  - The output returned by the function $F$ for a given input $x_0$ is to be a given number $y_0$.

**Example 16.** Find the global input-output rule of the function $KATE$ given that it is affine, that its slope is $-3$ and that the output for the input $+2$ is $+5$.

We use all three given pieces of information:

i. Since we are given that $KATE$ is an affine function, we give temporary names for the dilation coefficient, say $a$, and for the constant term, say $b$, and we write the global input-output rule of $KATE$ in terms of these names:

\[x \xrightarrow{KATE_{a,b}} KATE_{a,b}(x) = ax + b\]
ii. By the **Local Slope Theorem**, the slope is equal to the dilation coefficient:

\[-3 = a\]

which give the equation \(a = -3\)

iii. Since the output for the input +2 is +5, we write

\[KATE_{a,b}(x)|_{x=+2} = +5\]

\[ax + b|_{x=+2} = +5\]

\[a(+2) + b = +5\]

which give the equation \(2a + b = +5\)

iv. So we must solve the system of two equations for two unknowns \(a\) and \(b\):

\[\begin{align*}
    a &= -3 \\
    2a + b &= +5
\end{align*}\]

This kind of system is very simple to solve since we need only replace \(a\) by \(-3\) in the second equation to get the equation:

\[2(-3) + b = +5\]

which we solve using the **Reduction Method**:

\[-6 + b = +5\]

\[-6 + b + 6 = +5 + 6\]

\[b = +11\]

v. So, the global input-output rule for \(KATE\) is

\[x \xrightarrow{KATE_{-3,+11}} KATE_{-3,+11}(x) = -3x + 11\]

### 10.14 Boundary Value Problem

A **Boundary Value Problem** asks the question:

- What is the *input-output rule* of a function \(F\), given that:
  - The function \(F\) is affine
  - The output returned by the function \(F\) for a given input \(x_1\) is to be a given number \(y_1\).
  - The output returned by the function \(F\) for a given input \(x_2\) is to be a given number \(y_2\).

**Example 17.** Find the global input-output rule of the function \(DAVE\) given that it is affine, that the output for the input +2 is −1 and that the output for the input −4 is −19.

We use all three pieces of information that we are given:
i. Since we are given that $DAVE$ is an affine function, we give temporary names for the dilation coefficient, say $a$, and for the constant term, say $b$, and we write the global input-output rule of $DAVE$ in terms of these names:

$$x \xrightarrow{DAVE_{a,b}} DAVEa, b(x) = ax + b$$

ii. Since the output for the input $+2$ is $-1$ we write:

$$DAVE_{a,b}(x) ig|_{x:=+2} = -1$$

$$ax + b ig|_{x:=+2} = -1$$

which give the equation $+2a + b = -1$

iii. Since the output for the input $-4$ is $-19$ we write:

$$DAVE_{a,b}(x) ig|_{x:=-4} = -19$$

$$ax + b ig|_{x:=-4} = -19$$

which give the equation $-4a + b = -19$

iv. So we must solve the system of two equations for two unknowns $a$ and $b$:

$$\begin{cases} +2a + b = -1 \\ -4a + b = -19 \end{cases}$$

This kind of system is a bit more complicated to solve but since $b$ appears in both equations, we replace one of the two equations, say the second one, by “the first one minus the second one”:

$$\begin{cases} +2a + b = -1 \\ (+2a + b) - [-4a + b] = [-1] - [-19] \end{cases}$$

This gives us:

$$\begin{cases} +2a + b = -1 \\ +2a + b + 4a - b = -1 + 19 \end{cases}$$

that is

$$\begin{cases} +2a + b = -1 \\ +6a = +18 \end{cases}$$

that is

$$\begin{cases} +2a + b = -1 \\ -6a = -18 \\ +6 = +6 \end{cases}$$

that is

$$\begin{cases} +2a + b = -1 \\ a = +3 \end{cases}$$

and now we replace in the first equation $a$ by $+3$: 
that is
\[
\begin{cases}
+2a + b = -1 |_{a=+3} \\
a = +3
\end{cases}
\]
that is
\[
\begin{cases}
+2(+3) + b = -1 \\
a = +3
\end{cases}
\]
and we reduce the first equation
\[
\begin{cases}
+6 + b - 6 = -1 - 6 \\
a = +3
\end{cases}
\]
which gives us, finally
\[
\begin{cases}
b = -7 \\
a = +3
\end{cases}
\]
\textbf{v.} So the global input-output rule of } DAVE \text{ is}
\[
x \xrightarrow{DAVE_{+3,-7}} DAVE_{+3,-7}(x) = +3x - 7
\]