

***On the Notion of Constructive Sets and Function Spaces
in the Framework of Markov's Mathematio-Logic***

***О понятии конструктивных множеств и функциональных пространств
в рамках математико-логики Маркова***

論馬可夫數理邏輯架構中的建構集合與函數空間的概念

Krishnan V.

*Department of Mathematics, Jamal Mohamed College, Tiruchirappalli
vkmaths@jmc.edu*

Rajan E.G.

*Professor of Signal Processing, Chairman, Pentagram Group of Companies
dr.rajaneg@gmail.com*

The concept of a set is not unique in constructive mathematics, for, it depends on the language $\mathcal{Y}_\alpha$, $0 \leq \alpha \leq \omega$, chosen to interpret the concept and hence it varies from case to case. In loose terms, a constructive set consists only of constructive objects. A mathematical object of analysis is known as a constructive object when it is represented as a word from some alphabet. The terms set and property are considered here to be synonymous. A constructive object is said to be a member of a set if it exhibits the corresponding property.

Let X_0 denote a collection of some words from an alphabet A . Let P_0 be some property satisfied by the elements of X_0 . Then the property P_0 is said to be potentially enumerable on X_0 if one could specify another property P_1 which holds for those and only those elements of X_0 . Given an alphabet A , if a variable s ranges over the entire free monoid A^* then s is known as a base variable. On the other hand, restricted variables are defined as those variables whose admissible values (words) are from a finite collection of words from A . Now let us consider a one-parameter formula ϕ_α whose parameter is an individual base variable. Here, ϕ_α is simply written as ϕ . ϕ is called an algorithmically verifiable condition (recursive condition), if one can construct some normal algorithm which would verify the fulfillment of the condition imposed by ϕ . Let ϕ be a formula of the language $\mathcal{Y}_\omega$ of the type $\forall x B(x)$ where $B(x)$ is a normal formula and x is a restricted variable whose admissible values are from some finite collection, say, X of words from a specific alphabet A . Then one can construct a normal algorithm N^ϵ which transforms every word chosen by x from X into O if $B(x)$ holds or into 0 if $B(x)$ does not hold. Then X is said to be decided by the algorithm N^ϵ . In other words, given an alphabet A and a property P , one can construct a normal algorithm N^ϵ over A which would decide the set X of those constructive objects from A satisfying the property P . Let us denote for convenience, this normal algorithm by N_X^ϵ is viewed as the preset deciding process. A preset might

contain more than one identical constructive objects which satisfy the stipulated property whereas such a repetition is not allowed in the notion of a set. In order to overcome this difficulty, one can construct a normal algorithm $N_X \dot{=}$ which would regulate a preset X to contain only unique and nonrepetitive constructive objects, by operating on every pair of elements of X , say, y and z represented as $y * z$ in the following manner: if $y \dot{=} z$, then $N_X \dot{=} (y * z) = y$ and If $y \neq z$, then $N_X \dot{=}$ is not sa-definite for $y * z$ and the pair $\langle y, z \rangle$ remains unaltered. A normal algorithm $N_X \dot{=}$ is defined as a regular of uniqueness in itself of a preset deciding process N_X^ϵ if for every y and z from X the following holds: $\forall yz \& \supset (y \dot{=} z) \rightarrow N_X \dot{=} \lfloor y * z \rfloor \dot{=} y \supset (y \neq z) \rightarrow \neg! N_X \dot{=} \lfloor y * z \rfloor$. Based on certain ideas discussed earlier, we provide below a technique using which one can construct a normal algorithm of the type $N_X \dot{=}$.

By a constructive set we mean the canonically represented verboid of the form $\{N_X^\epsilon\} \sqsubseteq \{N_X \dot{=}\}$ from the alphabet $A_{15} = \{o \mid \}, \{ \sqsubseteq \}$ where, $\{N_X^\epsilon\}$ decides the preset by identifying the verboidal transcriptions of those constructive objects which satisfy the property P stipulated by a one-parameter formula $B(x)$ and $\{N_X \dot{=}\}$ is the corresponding regulator of uniqueness in itself of $\{N_X^\epsilon\}$. Without loss of generality, we can interpret a set process $\{N_X^\epsilon\} \sqsubseteq \{N_X \dot{=}\}$ as $\{\mathfrak{R}_X^\epsilon\} \sqsubseteq \{\mathfrak{R}_X \dot{=}\}$ so as to generalize the notion of a constructive set as the one which is decided by a part of constructive system $\mathfrak{R}_X^\epsilon$ and $\mathfrak{R}_X \dot{=}$. The successful termination of the set process $\{\mathfrak{R}_X^\epsilon\} \sqsubseteq \mathfrak{R}_X \dot{=}$ yields the desired set X which is the string of the form $\{X\uparrow\}$, where the term $X\uparrow$, is the $\dot{=}$ -dilution of all the unique objects decided by the process. Let P be a potentially enumerable property (PEP) of a set X constructed by $\{\mathfrak{R}_X^\epsilon\} \sqsubseteq \{\mathfrak{R}_X \dot{=}\}$. Then X is said to be finite if the set process $\{\mathfrak{R}_X^\epsilon\} \sqsubseteq \{\mathfrak{R}_X \dot{=}\}$ terminates. On the other hand, X is said to be nonfinite if the set process is proved to be nonterminating. X is said to be quasifinite, if one fails to prove that the set process does not terminate. By the complement of a set X words from an alphabet A with respect to the entire set of all words from A , we mean the set $\bar{X}$ of words from A defined by the condition $P \in \bar{X} \equiv \neg(P \in X)$ such that P does not satisfy the corresponding PEP of the set X .

One can define various notions pertaining to sets and function spaces using the notion of constructive set described in this paper.

Понятие множества не уникально в конструктивной математике, поскольку оно зависит от языка $\mathcal{Y}_\alpha$, $0 \leq \alpha \leq \omega$, выбранного для интерпретации этого понятия, и, следовательно, варьируется от случая к случаю. Грубо говоря, конструктивное множество состоит только из конструктивных объектов.

Математический объект анализа называется конструктивным, если он представлен в виде слова из некоторого алфавита. Термины множество и свойство рассматриваются здесь как синонимы. Конструктивный объект называется членом множества, если он обладает соответствующим свойством.

Пусть X_0 обозначает набор некоторых слов из алфавита A . Пусть P_0 — некоторое свойство, которому удовлетворяют элементы X_0 . Тогда свойство P_0 называется потенциально перечислимым на X_0 , если можно указать другое свойство P_1 , которое справедливо для тех и только этих элементов X_0 . Учитывая алфавит A , если переменная s охватывает весь свободный моноид A^* , то s называется базовой переменной. С другой стороны, ограниченные переменные определяются как те переменные, допустимые значения (слова) которых взяты из конечного набора слов из A . Теперь рассмотрим однопараметрическую формулу $\phi\alpha$, параметром которой является отдельная базовая переменная. Здесь $\phi\alpha$ просто пишется как ϕ . ϕ называется алгоритмически проверяемым условием (рекурсивным условием), если можно построить некоторый нормальный алгоритм, проверяющий выполнение условия, налагаемого ϕ . Пусть ϕ — формула языка $\mathcal{Y}_0$ типа $\forall x B(x)$, где $B(x)$ — нормальная формула, а x — ограниченная переменная, допустимые значения которой взяты из некоторого конечного набора, скажем, X слов из определенного алфавита A . Тогда можно построить нормальный алгоритм N^ϵ , который преобразует каждое слово, выбранное x , из X в O , если $B(x)$ выполнено, или в 0 если $B(x)$ не выполняется. Тогда говорят, что X определяется алгоритмом N^ϵ . Другими словами, учитывая алфавит A и свойство P , можно построить нормальный алгоритм N^ϵ над A , который будет определять множество X тех конструктивных объектов из A , которые удовлетворяют свойству P . Обозначим для удобства этот нормальный алгоритм через N^{X^ϵ} рассматривается как предустановленный процесс принятия решения. Набор может содержать более одного одинаковых конструктивных объектов, удовлетворяющих оговоренному свойству, тогда как в понятии набора такое повторение не допускается. Чтобы преодолеть эту трудность, можно построить нормальный алгоритм $N_X^\cdot$, который бы регулировал предустановку X так, чтобы она содержала только уникальные и неповторяющиеся конструктивные объекты, оперируя каждой парой элементов X , скажем, y и z , представленных как $y*z$ в следующем образом: если $y \stackrel{\cdot}{=} z$, то $N_X^\cdot (y * z) = y$ и если $y \stackrel{\cdot}{\neq} z$, то $N_X^\cdot$ не является sa -определенным для $y*z$ и пара $\langle y, z \rangle$ остается неизменной. Нормальный алгоритм $N_X^\cdot$ определяется как регулярный показатель уникальности заранее заданного решающего процесса N^{X^ϵ} , если для любых y и z из X выполняется следующее: $\forall yz \& \supset (y \stackrel{\cdot}{=} z) \rightarrow N_X^\cdot \lfloor y * z \rfloor \stackrel{\cdot}{=} y \supset (y \stackrel{\cdot}{\neq} z) \rightarrow \neg \lfloor y * z \rfloor$. Основываясь на некоторых идеях, обсуждавшихся ранее, ниже мы предлагаем технику, с помощью которой можно

построить нормальный алгоритм типа $N_X \dot{=}$.

Под конструктивным множеством будем понимать канонически представленный вербоид вида $\{N_X \in\} \sqcap \{N_X \dot{=}\}$ из алфавита $A15 = \{o | \}, \{ \sqcap \}$, где $\{N_X \in\}$ определяет предустановку путем идентификации вербоидных транскрипций тех конструктивных объектов, которые удовлетворяют свойству P , обусловленному однопараметрической формулой $B(x)$, а $\{N_X \dot{=}\}$ является соответствующим регулятором уникальности в сам $\{N_X \in\}$. Без потери общности мы можем интерпретировать процесс набора $\{N_X \in\} \sqcap \{N_X \dot{=}\}$ как $\{\mathcal{R}_X \in\} \sqcap \mathcal{R}_X \dot{=}$, чтобы обобщить понятие конструктивного множества как набора, который определяется частью конструктивной системы. $\mathcal{R}_X \in$ и $\mathcal{R}_X \dot{=}$. Успешное завершение процесса набора $\{\mathcal{R}_X \in\} \sqcap \{\mathcal{R}_X \dot{=}\}$ дает желаемый набор X , который представляет собой строку вида $\{X^\dagger, \}$, где термин $X^\dagger$ представляет собой ,-разбавление всех уникальные объекты, определенные в процессе. Пусть P — потенциально перечислимое свойство (PER) множества X , построенное по $\{\mathcal{R}_X \in\} \sqcap \{\mathcal{R}_X \dot{=}\}$. Тогда X называется конечным, если процесс множества $\{\mathcal{R}_X \in\} \sqcap \{\mathcal{R}_X \dot{=}\}$ завершается. С другой стороны, X называется бесконечным, если доказано, что процесс множества не завершается. X называется квазиконечным, если не удастся доказать, что заданный процесс не завершается. Под дополнением множества X слов из алфавита A по отношению ко всему множеству всех слов из A будем понимать множество $\bar{X}$ слов из A , определенное условием $P \in \bar{X} \equiv \neg(P \in X)$ такое, что P не удовлетворяет соответствующему PER множества X .

Можно определить различные понятия, относящиеся к множествам и функциональным пространствам, используя понятие конструктивного множества, описанное в этой статье.

集合的概念在構造性數學中並不是唯一的，因為它取決於用來解釋該概念的語言 $\mathcal{Y}_\omega, 0 \leq \alpha \leq \omega$ ，因此它會因情況而異。寬鬆地說，構造性集合僅由構造性物件組成。當數學分析物件被表示為某個字母表中的單字時，它被稱為構造性物件。術語集和屬性在這裡被認為是同義的。如果構造物件表現出對應的屬性，則稱該物件是集合的成員。

令 X_0 表示字母表 A 中一些單字的集合。令 P_0 是 X_0 的元素滿足的某個屬性。那麼，如果可以指定另一個屬性 P_1 ，該屬性適用於且僅適用於 X_0 的那些元素，則屬性 P_0 被認為是 X_0 上潛在可枚舉的。給定字母表 A ，如果變數 s 的範圍涵蓋整個自由么半群 A^* ，則 s 稱為基底變數。另一方面，受限變數被定義為那些其允許值（單字）來自 A 的有限單字集合的變數。現在讓我們考慮一個單參數公式 ϕ_α ，其參數是單一基底變數。這裡， ϕ_α 簡稱為 ϕ 。

如果可以建構某種正常演算法來驗證 f 所施加的條件是否滿足，則 f 稱為演算法上可驗證的條件（遞歸條件）。設 f 為語言 $\mathcal{A}_0$ 的公式類型為 $\forall x B(x)$ ，其中 $B(x)$ 是正規公式， x 是受限變量，其允許值來自某個有限集合，例如來自特定字母表 A 的單字 X 。然後可以構造正規演算法 $N \in$ 如果 $B(x)$ 成立，則將 x 從 X 選擇的每個單字轉換為 0 或轉換為 0 如果 $B(x)$ 不成立。那麼 X 據說是由演算法 $N \in$ 決定的。換句話說，給定一個字母表 A 和一個屬性 P ，我們可以在 A 上建立一個正規演算法 $N \in$ ，它將從 A 中決定那些滿足屬性 P 的建構物件的集合 X 。為了方便起見，我們將這個正規演算法表示為 $N_X \in$ 被視為預設的決策過程。預設可能包含多個滿足規定屬性的相同構造對象，而集合概念中不允許這種重複。為了克服這個困難，我們可以建立一個正常的演算法 $N_X \in$ ，透過對 X 的每一對元素（例如表示為 $y * z$ 的 y 和 z ）進行操作，將預設的 X 調整為僅包含唯一且非重複的構造物件。如下方式：如果 $y \in z$ ，則 $N_X \in (y = z) = y$ 並且如果 $y \neq z$ ，則 $N_X \in$ 對於 $y * z$ 不是 sa 定的，並且 $\langle y, z \rangle$ 對保持不變。正常演算法 $N_X \in$ 定義為預設決策過程 $N_X \in$ 本身唯一性的正則，如果對於 X 中的每個 y 和 z 滿足以下條件： $\forall yz \supset (y \in z) \rightarrow N_X \in [y * z] \in y \supset (y \neq z) \rightarrow N_X \in [y * z]$ 。基於前面討論的某些想法，我們在下面提供了一種技術，使用該技術可以建立 $N_X \in$ 類型的正常演算法。我們所說的構造集合是指來自字母表 $A_{15} = \{0 \mid\}$ 的形式為 $\{N_X \in\} \sqcap \{N_X \in\}$ 的規範所表示的動詞。 $A_{15} = \{0 \mid\}$ ， $\{\sqcap\}$ 其中， $\{N_X \in\}$ 透過辨識那些符合單參數公式 $B(x)$ 規定的屬性 P 的建構物的動詞轉錄來決定預設，而 $\{N_X \in\}$ 是唯一對應的性調節器 $\{N_X \in\}$ 本身。不失一般性，我們可以將集合過程 $\{N_X \in\} \sqcap \{N_X \in\}$ 解釋為 $\{\mathcal{R}_X \in\} \sqcap \{\mathcal{R}_X \in\}$ ，從而將構造型集合的概念推廣為由構造型系統的一部分決定的集合過程 $\mathcal{R}_X \in$ 和 $\mathcal{R}_X \in$ 。集合過程 $\{\mathcal{R}_X \in\} \sqcap \mathcal{R}_X \in$ 的成功終止產生所需的集合 X ，它是形式為 $\{X^\dagger, \}$ 的字串，其中項 $X^\dagger$ ，是所有由過程決定的獨特物件。令 P 為由 $\{\mathcal{R}_X \in\} \sqcap \{\mathcal{R}_X \in\}$ 所建構的集合 X 的潛在可列舉屬性（PEP）。若集合程序 $\{\mathcal{R}_X \in\} \sqcap \{\mathcal{R}_X \in\}$ 終止，則稱 X 為有限的。另一方面，如果證明集合過程是非終止的，則稱 X 是非有限的。如果無法證明集合過程不會終止，則稱 X 是擬有限的。透過字母 A 中的一組 X 個單字相對於 A 中所有單字的整個集合的補集，我們指的是由條件 $P \in \bar{X} \equiv \neg(P \in X)$ 定義的 A 中單字的集合 $\bar{X}$ ，使得 P 不滿足集合 X 的相應 PEP。

人們可以使用本文所描述的構造型集合的概念來定義與集合和函數空間相關的各種概念。