

DATA TRANSMISSION METHOD, ELECTRONIC DEVICE, AND COMPUTER READABLE MEDIUM

5

TECHNICAL FIELD

[0001] The present application relates to the field of communication technologies, in particular to a data transmission method, an electronic device, and a computer readable medium.

10

BACKGROUND

[0002] Long Term Evolution (LTE) technology is a wireless cellular communication technology of the 4th Generation mobile communication technology (4G). The LTE adopts Orthogonal Frequency Division Multiplexing (OFDM) technology, and time-frequency resources composed of subcarriers and OFDM symbols constitute wireless physical time-frequency resources of an LTE system. Currently, the OFDM technology has been widely applied in wireless communications. Due to the adoption of the Cyclic Prefix (CP), a CP-OFDM system can effectively solve a problem of multipath delay and convert a frequency-selective channel into a set of parallel flat-fading channels, which simplifies a channel estimation method and maintains a relatively high accuracy of channel estimation. However, the performance of the CP-OFDM system is sensitive to a frequency offset and a time offset between adjacent subbands, which is mainly due to a relatively large spectral leakage of the system and therefore prone to cause inter-subband interference. The LTE system currently uses a guard interval in frequency domain, which reduces spectrum efficiency.

25 [0003] The 5th Generation New Radio (5G NR) still adopts CP-OFDM as a basic waveform, and different parameter values such as subcarrier spacing and the like may be adopted by two adjacent subbands, which may destroy orthogonality between subcarriers and bring new problem of interference.

SUMMARY

[0004] Embodiments of the present application provide a data transmission method, an electronic device, and a computer readable medium.

5 [0005] In a first aspect, an embodiment of the present application provides a data transmission method, including: performing orthogonal spreading code transformation on data contained in first sequences satisfying a condition, in N groups of first sequences obtained by dividing a first data sequence to be transmitted, to generate N groups of second sequences; the N groups of first sequences correspond to N time-frequency resource blocks, and the first sequences satisfying the condition include the first sequences, 10 in the N groups of first sequences, corresponding to the time-frequency resource blocks with a subcarrier spacing different from a subcarrier spacing of adjacent time-frequency resource blocks; N being an integer greater than or equal to 2; performing first inverse Fourier transform with A-fold oversampling on the N groups of second sequences to obtain N groups of third sequences; A being an integer greater than or equal to 1; 15 performing second inverse Fourier transform on the N groups of third sequences to obtain a second data sequence; and transmitting the second data sequence on the N time-frequency resource blocks.

[0006] In a second aspect, an embodiment of the present application provides an electronic device, including: at least one processor; a memory having at least one program stored 20 thereon, the at least one program, executed by the at least one processor, implements the data transmission method described above.

[0007] In a third aspect, an embodiment of the present application provides a computer readable medium having a computer program stored thereon, the computer program, executed by a processor, implements the data transmission method described above.

25 [0008] In the data transmission method provided in the present application, the orthogonal spreading code transformation is performed on the data contained in the first sequences satisfying the condition, since performing orthogonal spreading code transformation on the data does not change the bandwidth occupied by the data, the interference between the time-frequency resource blocks corresponding to the first sequences satisfying the 30 condition is reduced without changing the bandwidth occupied by the first data sequence to be transmitted, so that spectrum efficiency is improved; and the N groups of second sequences obtained after the orthogonal spreading code transformation are then subjected

to inverse Fourier inverse transform twice, thereby facilitating subsequent filtering, resulting in a reduced complexity and a reduced processing time delay.

BRIEF DESCRIPTION OF DRAWINGS

- 5 [0009] Fig. 1 is a flowchart of a data transmission method provided in an embodiment of the present application;
- [0010] Fig. 2 is a schematic diagram of a processing procedure of processing a first data sequence to obtain a second data sequence in Example 1 of an embodiment of the present application;
- 10 [0011] Fig. 3 is a schematic diagram of a processing procedure of processing a first data sequence to obtain a second data sequence in Example 2 of an embodiment of the present application;
- [0012] Fig. 4 is a schematic diagram of a processing procedure of processing a first data sequence to obtain a second data sequence in Example 3 of an embodiment of the present application;
- 15 [0013] Fig. 5 is a schematic diagram of a processing procedure of processing a first data sequence to obtain a second data sequence in Example 4 of an embodiment of the present application;
- [0014] Fig. 6 is a schematic diagram of a processing procedure of processing a first data sequence to obtain a second data sequence in Example 5 of an embodiment of the present application;
- 20 [0015] Fig. 7 is a schematic diagram of a processing procedure of processing a first data sequence to obtain a second data sequence in Example 6 of an embodiment of the present application;
- [0016] Fig. 8 is a schematic diagram of a processing procedure of processing a first data sequence to obtain a second data sequence in Example 7 of an embodiment of the present application;
- 25 [0017] Fig. 9 is a block diagram of an electronic device provided in an embodiment of the present application.

30

DETAIL DESCRIPTION OF EMBODIMENTS

[0018] In order to make those skilled in the art better understand the technical solutions of the present application, a data transmission method, an electronic device, and a computer readable medium, provided in the present application, are described in detail below with reference to the accompanying drawings.

[0019] Exemplary embodiments are described in more detail below with reference to the accompanying drawings, but the exemplary embodiments may be embodied in different forms, and should not be explained as limited to the embodiments set forth herein. In contrast, the embodiments are provided to make the present application thorough and complete, and for those skilled in the art fully understanding the scope of the present application.

[0020] The embodiments of the present application and the technical features in the embodiments may be combined with each other if no conflict is incurred.

[0021] As used in the present application, a term "and/or" includes any and all combinations of at least one of listed items.

[0022] The terms used in the present application are for a purpose of describing particular embodiments only, and are not intended to limit the present application. As used in the present application, singular forms "a" and "the" are intended to include plural forms as well, unless the context clearly defines otherwise. It should further be understood that terms "includes/comprises (including/comprising)" and/or "made of/consisted of" in the present application are used to specify a presence of at least one of recited features, integers, steps, operations, elements or components, but do not preclude a presence or an addition of at least one of other features, integers, steps, operations, elements, components or groups thereof.

[0023] Unless otherwise defined, meanings of all terms (including technical terms and scientific terms) used herein are the same as meanings commonly understood by one of ordinary skills in the art. It should further be understood that terms, such as those defined in common dictionaries, should be construed as having a meaning that is consistent with that in background of the existing art and the present application, and should not be construed as having an idealized or over-formal meaning, unless expressly defined in the present application.

[0024] Fig. 1 is a flowchart of a data transmission method provided in an embodiment of

the present application.

[0025] In a first aspect, referring to Fig. 1, an embodiment of the present application provides a data transmission method, including an operation 100.

5 [0026] At operation 100, performing orthogonal spreading code transformation on data contained in first sequences satisfying a condition, in N groups of first sequences obtained by dividing a first data sequence to be transmitted, to generate N groups of second sequences; the N groups of first sequences correspond to N time-frequency resource blocks, and the first sequences satisfying the condition include the first sequences, in the N groups of first sequences, corresponding to the time-frequency resource blocks with a
10 subcarrier spacing different from that of adjacent time-frequency resource blocks; N being an integer greater than or equal to 2.

[0027] In some implementations, the first data sequence to be transmitted includes constellation point-modulated data and R pieces of reference signal data, R being an integer greater than or equal to 0.

15 [0028] In some implementations, the first data sequence to be transmitted is part or all of data to be transmitted in a channel bandwidth.

[0029] In some implementations, the N time-frequency resource blocks are part or all of time-frequency resource blocks in the channel bandwidth.

20 [0030] In some implementations, each of the N time-frequency resource blocks corresponds to one subband.

[0031] In some implementations, an n^{th} group of first sequences corresponds to an n^{th} time-frequency resource block; n being an integer greater than or equal to 1 and less than or equal to N.

25 [0032] In some implementations, an n^{th} time-frequency resource block corresponding to an n^{th} group of first sequences refers to that data contained in the n^{th} group of first sequences are transmitted on the n^{th} time-frequency resource block.

[0033] In some implementations, in the N time-frequency resource blocks, a subcarrier spacing between any adjacent subcarriers contained in any one time-frequency resource block is the same.

30 [0034] In some implementations, the time-frequency resource blocks with a same subcarrier spacing each contain a same number of subcarriers.

[0035] In some implementations, the time-frequency resource blocks with different subcarrier spacings contain different numbers of subcarriers.

[0036] In some implementations, in the N time-frequency resource blocks, there are at least two time-frequency resource blocks with different subcarrier spacings. For example, there are at least an a^{th} time-frequency resource block and a b^{th} time-frequency resource block, with a subcarrier spacing between subcarriers contained in the a^{th} time-frequency resource block being different from a subcarrier spacing between subcarriers contained in the b^{th} time-frequency resource block; a and b being integers greater than or equal to 1 and less than or equal to N.

[0037] In some implementations, in the N time-frequency resource blocks, there are at least two adjacent time-frequency resource blocks with different subcarrier spacings. For example, there are at least an a^{th} time-frequency resource block and an $(a+1)^{\text{th}}$ time-frequency resource block, with a subcarrier spacing between subcarriers contained in the a^{th} time-frequency resource block being different from a subcarrier spacing between subcarriers contained in the $(a+1)^{\text{th}}$ time-frequency resource block.

[0038] In some implementations, in order to reduce interference between the time-frequency resource blocks corresponding to a first group of first sequences and an N^{th} group of first sequences and other time-frequency resource blocks except the N time-frequency resource blocks corresponding to the N groups of first sequences, the first sequences satisfying the condition further include: the first group of first sequences and the N^{th} group of first sequences.

[0039] In some implementations, a total number of the groups of first sequences satisfying the condition is greater than or equal to 2, and less than or equal to N.

[0040] In some implementations, a subcarrier spacing of a time-frequency resource block corresponding to the first sequences satisfying the condition is different from a subcarrier spacing of a previous time-frequency resource block.

[0041] In some implementations, the subcarrier spacing of the time-frequency resource block corresponding to the first sequences satisfying the condition is different from a subcarrier spacing of a next time-frequency resource block.

[0042] In some implementations, the subcarrier spacing of the time-frequency resource block corresponding to the first sequences satisfying the condition is different from the subcarrier spacings of the previous time-frequency resource block and the next time-frequency resource block.

[0043] In some implementations, a ratio of a total number of pieces of data contained in the first sequences satisfying the condition to a total number of pieces of data contained

in a previous group of first sequences is 2 to the power of i ; i being an integer greater than or equal to 1.

[0044] In some implementations, a ratio of the number of pieces of data contained in the first sequences satisfying the condition to a total number of pieces of data contained in a next group of first sequences is 2 to the power of i .

[0045] In some implementations, the ratio of the number of pieces of data contained in the first sequences satisfying the condition to the number of pieces of data contained in the previous group of first sequences is 2 to the power of i , and the ratio of the number of pieces of data contained in the first sequences satisfying the condition to the number of pieces of data contained in the next group of first sequences is 2 to the power of i .

[0046] For example, in a case where the subcarrier spacing of the time-frequency resource block corresponding to the first sequences satisfying the condition is different from the subcarrier spacing of the previous time-frequency resource block, the ratio of the number of pieces of data contained in the first sequences satisfying the condition to the number of pieces of data contained in the previous group of first sequences is 2 to the power of i ; in a case where the subcarrier spacing of the time-frequency resource block corresponding to the first sequences satisfying the condition is different from the subcarrier spacing of the next time-frequency resource block, the ratio of the number of pieces of data contained in the first sequences satisfying the condition to the number of pieces of data contained in the next group of first sequences is 2 to the power of i ; in a case where the subcarrier spacing of the time-frequency resource block corresponding to the first sequences satisfying the condition is different from the subcarrier spacings of the previous time-frequency resource block and the next time-frequency resource block, the ratio of the number of pieces of data contained in the first sequences satisfying the condition to the number of pieces of data contained in the previous group of first sequences is 2 to the power of i , and the ratio of the number of pieces of data contained in the first sequences satisfying the condition to the number of pieces of data contained in the next group of first sequences is 2 to the power of i .

[0047] In some implementations, a second sequence corresponding to each group of first sequences not satisfying the condition is the same as the first sequence.

[0048] In some implementations, the performing orthogonal spreading code transformation on data contained in first sequences satisfying a condition, in N groups of first sequences obtained by dividing a first data sequence to be transmitted, to generate N

groups of second sequences includes: performing orthogonal spreading code transformation on data to be transformed in the data contained in the first sequences satisfying the condition to obtain the second sequence, with the data to be transformed being fore part data, or rear part data, or both the fore part data and the rear part data, or all data, in the data contained in the first sequences satisfying the condition.

[0049] In some implementations, in a case where the subcarrier spacing of the time-frequency resource block corresponding to the first sequences satisfying the condition is different from the subcarrier spacing of the previous time-frequency resource block, the data to be transformed is the fore part data or all data; in a case where the subcarrier spacing of the time-frequency resource block corresponding to the first sequences satisfying the condition is different from the subcarrier spacing of the next time-frequency resource block, the data to be transformed is the rear part data or all data; in a case where the subcarrier spacing of the time-frequency resource block corresponding to the first sequences satisfying the condition is different from the subcarrier spacings of the previous time-frequency resource block and the next time-frequency resource block, the data to be transformed is the fore part data and the rear part data, or all data.

[0050] In some implementations, the fore part data is first h_1 pieces of data in the data contained in the first sequences satisfying the condition, the rear part data is last h_2 pieces of data in the data contained in the first sequences satisfying the condition; h_1 and h_2 are integers greater than or equal to 1, and less than or equal to the number of pieces of data contained in the first sequences satisfying the condition.

[0051] In some implementations, the data to be transformed excludes reference signal data.

[0052] In some implementations, in a case where the data to be transformed includes the reference signal data, the reference signal data in the data to be transformed is to be placed to other positions so that the data to be transformed does not include the reference signal data.

[0053] In some implementations, the data to be transformed includes h pieces of data, and data obtained by performing orthogonal spreading code transformation on the data to be transformed includes hq/p pieces of data; h being an integer greater than or equal to 1, p being a total number of groups of first expanding sequences used for the orthogonal spreading code transformation, and q being a length of the first expanding sequence.

[0054] In some implementations, the performing orthogonal spreading code transformation on data to be transformed in the data contained in the first sequences

satisfying the condition to obtain the second sequence includes: expanding p groups of eighth sequences, obtained by dividing the data to be transformed, into p groups of ninth sequences with a length of hq/p ; p being a total number of groups of first expanding sequences used for the orthogonal spreading code transformation, and q being a length of the first expanding sequence; expanding p groups of first expanding sequences with a length of q into p groups of second expanding sequences with a length of hq/p ; multiplying corresponding data in a k^{th} group of ninth sequences and a k^{th} group of second expanding sequences to obtain a k^{th} group of tenth sequences; k being an integer greater than or equal to 1, and less than or equal to p ; performing a weighted summation on p groups of tenth sequences to obtain an eleventh sequence; splicing the eleventh sequence with other data except the data to be transformed in the data contained in the first sequences satisfying the condition to obtain the second sequence.

[0055] In some implementations, the performing orthogonal spreading code transformation on data to be transformed in the data contained in the first sequences satisfying the condition to obtain the second sequence includes: multiplying each piece of data in a k^{th} group of eighth sequences in p groups of eighth sequences, obtained by dividing the data to be transformed, by each piece of data in a k^{th} group of first expanding sequences in p groups of first expanding sequences with a length of q to obtain a k^{th} group of tenth sequences; k being an integer greater than or equal to 1 and less than or equal to p ; performing a weighted summation on p groups of tenth sequences to obtain an eleventh sequence; splicing the eleventh sequence with other data except the data to be transformed in the data contained in the first sequences satisfying the condition to obtain the second sequence.

[0056] In some implementations, the first expanding sequence is an orthogonal spreading code.

[0057] In some implementations, p is any one of 1, 2 or 3, and q is 4.

[0058] In some implementations, in a case where p is 3 and q is 4, the p groups of first expanding sequences are $S1=[C, -C, -C, C]$, $S2=[C, -C, C, -C]$, $S3=[C, C, -C, -C]$; C being an integer greater than or equal to 1.

[0059] In some implementations, C is 1.

[0060] In some implementations, the data to be transformed is divided into p groups of eighth sequences, each group of eighth sequences including h/p pieces of data, h/p being an integer.

[0061] In some implementations, the expanding p groups of eighth sequences, obtained by dividing the data to be transformed, into p groups of ninth sequences with a length of hq/p includes: for each group of eighth sequences, repeating each piece of data in the eighth sequences q times, and splicing the pieces of data each repeated q times together to obtain the ninth sequence with the length of hq/p . For example, a certain group of eighth sequences is $[a_1, a_2, a_3, a_4]$, and q is 4, then the ninth sequence corresponding to the group of eighth sequences is $[a_1, a_1, a_1, a_1, a_2, a_2, a_2, a_2, a_3, a_3, a_3, a_3, a_4, a_4, a_4, a_4]$.

[0062] In some implementations, the expanding p groups of first expanding sequences with a length of q into p groups of second expanding sequences with a length of hq/p includes: for each group of first expanding sequences, repeating all data in the first expanding sequences h/p times to obtain the second expanding sequence. For example, a certain group of first expanding sequences is $[C, -C, -C, C]$, and h/p is 4, then the second expanding sequence is $[C, -C, -C, C, C, -C, -C, C, C, -C, -C, C, C, -C, -C, C]$.

[0063] In some implementations, the multiplying corresponding data in a k^{th} group of ninth sequences and a k^{th} group of second expanding sequences to obtain a k^{th} group of tenth sequences refers to: multiplying data with a same position index in the k^{th} group of ninth sequences and the k^{th} group of second expanding sequences to obtain the k^{th} group of tenth sequences. For example, a first piece of data in the k^{th} group of ninth sequences and a first piece of data in the k^{th} group of second expanding sequences are multiplied to obtain a first piece of data in the k^{th} group of tenth sequences, a second piece of data in the k^{th} group of ninth sequences and a second piece of data in the k^{th} group of second expanding sequences are multiplied to obtain a second piece of data in the k^{th} group of tenth sequences, and so on.

[0064] In some implementations, the multiplying each piece of data in a k^{th} group of eighth sequences in p groups of eighth sequences, obtained by dividing the data to be transformed, by each piece of data in a k^{th} group of first expanding sequences in p groups of first expanding sequences with a length of q to obtain a k^{th} group of tenth sequences refers to: multiplying a first piece of data in the k^{th} group of eighth sequences by a first piece of data in the k^{th} group of first expanding sequences to obtain a first piece of data in the k^{th} group of tenth sequences, multiplying the first piece of data in the k^{th} group of eighth sequences by a second piece of data in the k^{th} group of first expanding sequences to obtain a second piece of data in the k^{th} group of tenth

sequences, and so on, until multiplying the first piece of data in the k^{th} group of eighth sequences by a q^{th} piece of data in the k^{th} group of first expanding sequences to obtain a q^{th} piece of data in the k^{th} group of tenth sequences; multiplying a second piece of data in the k^{th} group of eighth sequences by the first piece of data in the k^{th} group of first expanding sequences to obtain a $(q+1)^{\text{th}}$ piece of data in the k^{th} group of tenth sequences, multiplying the second piece of data in the k^{th} group of eighth sequences by the second piece of data in the k^{th} group of first expanding sequences to obtain a $(q+2)^{\text{th}}$ piece of data in the k^{th} group of tenth sequences, and so on, until multiplying the second piece of data in the k^{th} group of eighth sequences by the q^{th} piece of data in the k^{th} group of first expanding sequences to obtain a $(2q)^{\text{th}}$ piece of data in the k^{th} group of tenth sequences; and so on, until multiplying a $(h/p)^{\text{th}}$ piece of data in the k^{th} group of eighth sequences by the q^{th} piece of data in the k^{th} group of first expanding sequences to obtain a $(hq/p)^{\text{th}}$ piece of data in the k^{th} group of tenth sequences; finally, the k^{th} group of tenth sequences is obtained.

[0065] At operation 101, performing first inverse Fourier transform with A-fold oversampling on the N groups of second sequences to obtain N groups of third sequences; A being an integer greater than or equal to 1.

[0066] In some implementations, a total number of pieces of data contained in each group of second sequences is 2 to the power of i; i being an integer greater than or equal to 1.

[0067] In some implementations, a ratio of the number of pieces of data contained in any two groups of second sequences is 2 to the power of i; i being an integer greater than or equal to 1.

[0068] In some implementations, a total number of pieces of data contained in an n^{th} group of second sequences is $k'(n)$.

[0069] In some implementations, there are at least two groups of second sequences containing different numbers of pieces of data. For example, there are an $n1^{\text{th}}$ group and an $n2^{\text{th}}$ group of second sequences, with a total number of pieces of data contained in the $n1^{\text{th}}$ group of second sequences being different from a total number of pieces of data contained in the $n2^{\text{th}}$ group of second sequences; $n1$ and $n2$ being integers greater than or equal to 1, and less than or equal to N.

[0070] In some implementations, A is 2.

[0071] In some implementations, the performing first inverse Fourier transform with A-fold oversampling on the N groups of second sequences to obtain N groups of third

sequences includes: adding data of $\frac{k'(n)}{2}$ zeros at two ends of the n^{th} group of second sequences respectively, and performing first inverse Fourier transform on the n^{th} group of second sequences added with the data of zeros to obtain an n^{th} group of third sequences; n being an integer greater than or equal to 1 and less than or equal to N ,
5 and $k'(n)$ being a total number of pieces of data contained in the n^{th} group of second sequences.

[0072] In some implementations, a total number of inverse Fourier transform points for the first inverse Fourier transform performed on the n^{th} group of second sequences is the same as a total number of pieces of data contained in the n^{th} group of third
10 sequences; n being an integer greater than or equal to 1 and less than or equal to N .

[0073] In some implementations, a ratio of numbers of inverse Fourier transform points for the first inverse Fourier transform performed on any two groups of second sequences is 2 to the power of i ; i being an integer greater than or equal to 1.

[0074] In some implementations, there are at least two groups of second sequences, with
15 different numbers of inverse Fourier transform points for the first inverse Fourier transform performed thereon.

[0075] In some implementations, a total number of pieces of data contained in the n^{th} group of third sequences is $2k'(n)$.

[0076] In some implementations, a zero-frequency position of the first inverse Fourier
20 transform performed on the n^{th} group of second sequences is within a frequency range of the time-frequency resource block corresponding to the n^{th} group of first sequences; n being an integer greater than or equal to 1 and less than or equal to N .

[0077] In some implementations, zero-frequency positions of the first inverse Fourier transform performed on different groups of second sequences are different.

[0078] In some implementations, a zero-frequency position of the first inverse Fourier
25 transform performed on the n^{th} group of second sequences is in one of subcarriers contained in the time-frequency resource block corresponding to the n^{th} group of first sequences.

[0079] At operation 102, performing second inverse Fourier transform on the N groups
30 of third sequences to obtain a second data sequence.

[0080] In some implementations, a total number of pieces of data contained in the third sequence corresponding to the first sequences satisfying the condition is greater than

twice a total number of subcarriers contained in the time-frequency resource block corresponding to the first sequences satisfying the condition, a total number of pieces of data contained in the third sequence corresponding to the first sequences not satisfying the condition is twice a total number of subcarriers contained in the time-frequency resource block corresponding to the first sequences not satisfying the condition.

[0081] In some implementations, the third sequence corresponding to the first sequences satisfying the condition refers to a third sequence obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with A-fold oversampling on the first sequences satisfying the condition.

[0082] In some implementations, a total number of inverse Fourier transform points for the second inverse Fourier transform performed on the N groups of third sequences is greater than N.

[0083] In some implementations, in a case where the N groups of third sequences each contain a same number of pieces of data, the performing second inverse Fourier transform on the N groups of third sequences to obtain a second data sequence includes: performing second inverse Fourier transform with C-fold oversampling on N pieces of data consisting of a j^{th} data of the N groups of the third sequences to obtain a j^{th} group of fifth sequences; C being an integer greater than or equal to 1, j being an integer greater than or equal to 1, and less than or equal to B; expanding B groups of fifth sequences into B groups of sixth sequences; a total number of pieces of data contained in the sixth sequences is D times a total number of pieces of data contained in the fifth sequences; D being an integer greater than or equal to 2; splicing the B groups of sixth sequences into the second data sequence.

[0084] In some implementations, after expanding the B groups of fifth sequences into the B groups of sixth sequences, and before splicing the B groups of sixth sequences into the second data sequence, the method further includes: performing windowing operation on the B groups of sixth sequences.

[0085] The splicing the B groups of sixth sequences into the second data sequence includes: splicing the B groups of sixth sequences subjected to the windowing operation into the second data sequence.

[0086] In some implementations, the performing windowing operation on the B groups of sixth sequences refers to multiplying each group of sixth sequences by a preset

function.

[0087] In some implementations, the preset function is a window function.

[0088] In some implementations, the performing second inverse Fourier transform on the N groups of third sequences to obtain a second data sequence includes: performing
5 second inverse Fourier transform on the N groups of third sequences and other data sequences to obtain the second data sequence.

[0089] In some implementations, the other data sequences are data sequences except the first data sequence.

[0090] In some implementations, the other data sequences are data sequences obtained
10 by performing first inverse Fourier transform with A-fold oversampling on data sequences except the first data sequence.

[0091] In some implementations, the other data sequences are data sequences obtained by performing orthogonal spreading code transformation on data sequences except the first data sequence.

[0092] In some implementations, the other data sequences are data sequences obtained
15 by performing orthogonal spreading code transformation and first inverse Fourier transform with A-fold oversampling on data sequences except the first data sequence.

[0093] In some implementations, in a case where the N groups of third sequences and the other data sequences each contain a same number of pieces of data, the performing second
20 inverse Fourier transform on the N groups of third sequences and other data sequences to obtain the second data sequence includes: performing second inverse Fourier transform with C-fold oversampling on N+1 pieces of data consisting of a j^{th} piece of data of the N groups of third sequences and a j^{th} piece of data of the other data sequences to obtain a j^{th} group of fifth sequences; C being an integer greater than or
25 equal to 1, j being an integer greater than or equal to 1 and less than or equal to B; expanding B groups of fifth sequences into B groups of sixth sequences; a total number of pieces of data contained in the sixth sequences is D times a total number of pieces of data contained in the fifth sequences; D being an integer greater than or equal to 2; splicing the B groups of sixth sequences into the second data sequence.

[0094] In some implementations, after performing first inverse Fourier transform with A-
30 fold oversampling on the N groups of second sequences to obtain N groups of third sequences, and before performing second inverse Fourier transform on the N groups of third sequences to obtain the second data sequence, the method further includes:

expanding the N groups of third sequences into N groups of fourth sequences; a total number of pieces of data contained in each group of fourth sequences is B, B being a maximum value of a total number of pieces of data contained in the N groups of third sequences.

5 [0095] The performing second inverse Fourier transform on the N groups of third sequences to obtain a second data sequence includes: performing second inverse Fourier transform on the N groups of fourth sequences to obtain the second data sequence.

[0096] In some implementations, after performing first inverse Fourier transform with A-fold oversampling on the N groups of second sequences to obtain N groups of third
10 sequences, and before performing second inverse Fourier transform on the N groups of third sequences and other data sequences to obtain the second data sequence, the method further includes: expanding the N groups of third sequences and the other data sequences into N+1 groups of fourth sequences; a total number of pieces of data contained in each group of fourth sequences is B, B being a maximum value of a total number of
15 pieces of data contained in the N groups of third sequences.

[0097] The performing second inverse Fourier transform on the N groups of third sequences and other data sequences to obtain the second data sequence includes: performing second inverse Fourier transform on the N+1 groups of fourth sequences to obtain the second data sequence.

20 [0098] In some implementations, the expanding the N groups of third sequences into N groups of fourth sequences includes: in a case where a total number of pieces of data contained in a k^{th} group of third sequences is B, a k^{th} group of fourth sequences is the same as the k^{th} group of third sequences; in a case where the number of pieces of data contained in the k^{th} group of third sequences is less than B, splicing the k^{th} group of third
25 sequences with another k^{th} group of third sequences obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with A-fold oversampling on another first data sequence to be transmitted to obtain the k^{th} group of fourth sequences.

[0099] In some implementations, the expanding the N groups of third sequences and other
30 data sequences into N+1 groups of fourth sequences includes: in a case where a total number of pieces of data contained in a k^{th} group of third sequences is B, a k^{th} group of fourth sequences is the same as the k^{th} group of third sequences; in a case where the number of pieces of data contained in the k^{th} group of third sequences is less than B,

splicing the k^{th} group of third sequences with another k^{th} group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with A-fold oversampling on another first data sequence to be transmitted, to obtain the k^{th} group of fourth sequences.

- 5 [0100] In a case where a total number of pieces of data contained in the other data sequences is B, an $(N+1)^{\text{th}}$ group of fourth sequences is the same as the other data sequences; in a case where the number of pieces of data contained in the other data sequences is less than B, splicing the other data sequences with other data to obtain the $(N+1)^{\text{th}}$ group of fourth sequences.
- 10 [0101] In some implementations, the performing second inverse Fourier transform on the N groups of fourth sequences to obtain the second data sequence includes: performing second inverse Fourier transform with C-fold oversampling on N pieces of data consisting of a j^{th} piece of data of the N groups of fourth sequences to obtain a j^{th} group of fifth sequences; C being an integer greater than or equal to 1, j being an integer greater than or equal to 1, and less than or equal to B; expanding B groups of fifth sequences into B groups of sixth sequences; a total number of pieces of data contained in the sixth sequences is D times a total number of pieces of data contained in the fifth sequences; D being an integer greater than or equal to 2; splicing the B groups of sixth sequences into the second data sequence.
- 15 [0102] In some implementations, the performing second inverse Fourier transform on the N+1 groups of fourth sequences to obtain the second data sequence includes: performing second inverse Fourier transform with C-fold oversampling on N+1 pieces of data consisting of a j^{th} piece of data of the N+1 groups of fourth sequences to obtain a j^{th} group of fifth sequences; C being an integer greater than or equal to 1, j being an integer greater than or equal to 1, and less than or equal to B; expanding B groups of fifth sequences into B groups of sixth sequences; a total number of pieces of data contained in the sixth sequences is D times a total number of pieces of data contained in the fifth sequences; D being an integer greater than or equal to 2; splicing the B groups of sixth sequences into the second data sequence.
- 20 [0103] In some implementations, each group of fifth sequences corresponds to one sub-symbol.
- 25 [0104] In some implementations, the expanding B groups of fifth sequences into B groups of sixth sequences includes: repeating a k_1^{th} group of fifth sequences E times to
- 30

obtain a k_1^{th} group of sixth sequences; k_1 being an integer greater than or equal to 1 and less than or equal to B, E being an integer greater than or equal to 2.

[0105] In some implementations, E is 4.

5 [0106] In some implementations, the splicing the B groups of sixth sequences into the second data sequence refers to that, arranging the B groups of sixth sequences in a misaligned manner into B rows, a misaligned interval between a first piece of data in a $(k_1+1)^{\text{th}}$ row and a first piece of data in a k_1^{th} row is half of the number of pieces of data contained in the fifth sequences, and adding up data in a same column of the B rows to obtain the second data sequence.

10 [0107] In some implementations, during the B groups of sixth sequences being arranged in a misaligned manner into B rows, a sixth sequence arranged in a $(k_2)^{\text{th}}$ row is misaligned backward with respect to a sixth sequence arranged in a $(k_2-1)^{\text{th}}$ row, as shown in Fig. 8; k_2 being an integer greater than or equal to 2, and less than or equal to B.

15 [0108] In some implementations, after performing first inverse Fourier transform with A-fold oversampling on the N groups of second sequences to obtain N groups of third sequences, and before performing second inverse Fourier transform on the N groups of third sequences to obtain the second data sequence, the method further includes: performing coefficient correction on the N groups of third sequences to obtain N groups of seventh sequences.

20 [0109] The performing second inverse Fourier transform on the N groups of third sequences to obtain a second data sequence includes: performing second inverse Fourier transform on the N groups of seventh sequences.

[0110] In some implementations, the performing coefficient correction on the N groups of third sequences to obtain N groups of seventh sequences includes: multiplying even-
25 numbered groups of third sequences in the N groups of third sequences by corresponding data of a first coefficient vector to obtain the N groups of seventh sequences; or multiplying odd-numbered groups of third sequences in the N groups of third sequences by corresponding data of a second coefficient vector to obtain the N groups of seventh sequences.

30 [0111] In some implementations, the multiplying even-numbered groups of third sequences in the N groups of third sequences by corresponding data of a first coefficient vector to obtain the N groups of seventh sequences refers to, multiplying data with a same position index in the even-numbered groups of third sequences and the first coefficient

vector to obtain corresponding seventh sequences, and odd-numbered groups of third sequences being the same as corresponding seventh sequences.

[0112] In some implementations, the multiplying odd-numbered groups of third sequences in the N groups of third sequences by corresponding data of a second coefficient vector to obtain the N groups of seventh sequences refers to, multiplying data with a same position index in the odd-numbered groups of third sequences and the second coefficient vector to obtain corresponding seventh sequences, and even-numbered groups of third sequences being the same as corresponding seventh sequences.

[0113] In some implementations, the first coefficient vector is $[1, -1, 1, \dots, (-1)^{2k'(n)-1}]$.

[0114] In some implementations, the second coefficient vector is $[-1, 1, -1, \dots, (-1)^{2k'(n)}]$.

[0115] At operation 103, transmitting the second data sequence on the N time-frequency resource blocks.

[0116] In some implementations, after performing second inverse Fourier transform on the N groups of third sequences to obtain the second data sequence, and before transmitting the second data sequence on the N time-frequency resource blocks, the method further includes: filtering the second data sequence.

[0117] The transmitting the second data sequence on the N time-frequency resource blocks includes: transmitting filtered second data sequence on the N time-frequency resource blocks.

[0118] In some implementations, the filtering is single-phase filtering or polyphase filtering.

[0119] In some implementations, a filtering function used for polyphase filtering includes: a root raised cosine function, a raised cosine function, a rectangular function, an Isotropic Orthogonal Transform Algorithm (IOTA) function, or the like.

[0120] In the data transmission method provided in the present application, the orthogonal spreading code transformation is performed on the data contained in the first sequences satisfying the condition, since performing orthogonal spreading code transformation on the data does not change the bandwidth occupied by the data, the interference between the time-frequency resource blocks corresponding to the first sequences satisfying the condition is reduced without changing the bandwidth occupied by the first data sequence to be transmitted, so that spectrum efficiency is improved; and the N groups of second sequences obtained after the orthogonal spreading code transformation are then subjected to inverse Fourier inverse transform twice, thereby facilitating subsequent filtering,

resulting in a reduced complexity and a reduced processing time delay.

[0121] In order to provide a clearer illustration of a specific implementation process of the data transmission method provided in the embodiment of the present application, several examples are listed below for explanation, and are not used to limit the scope of protection for the embodiment of the present application.

[0122] Example 1

[0123] In the example, a first data sequence to be transmitted is divided into four groups of first sequences, a first group of first sequences contains 64 pieces of data, a second group of first sequences contains 60 pieces of data, a third group of first sequences contains 30 pieces of data, a fourth group of first sequences contains 32 pieces of data, as shown in Fig. 6.

[0124] The data in the first group and the second group of first sequences are transmitted on the time-frequency resource blocks with a subcarrier spacing of 15kHz, and the data in the third group and the fourth group of first sequences are transmitted on the time-frequency resource blocks with a subcarrier spacing of 30 kHz.

[0125] In order to suppress interference between the time-frequency resource blocks with different subcarrier spacings, orthogonal spreading code transformation is respectively performed on last 12 pieces of data in the data contained in the second group of first sequences and first 6 pieces of data in the data contained in the third group of first sequences, and a specific process is as follows.

[0126] First expanding sequences with $p=3$ and $q=4$ are selected, i.e., the three groups of first expanding sequences respectively are: $[1, -1, -1, 1]$, $[1, -1, 1, -1]$, $[1, 1, -1, -1]$.

[0127] For the second group of first sequences, the last 12 pieces of data are divided into three groups of eighth sequences, a first group of eighth sequences is $[a_1, a_2, a_3, a_4]$, a second group of eighth sequences is $[b_1, b_2, b_3, b_4]$, a third group of eighth sequences is $[c_1, c_2, c_3, c_4]$.

[0128] Each piece of data in the first group of eighth sequences $[a_1, a_2, a_3, a_4]$ is multiplied by the first group of first expanding sequences $[1, -1, -1, 1]$ and then spliced to obtain a first group of tenth sequences, i.e., $[a_1, -a_1, -a_1, a_1, a_2, -a_2, -a_2, a_2, a_3, -a_3, -a_3, a_3, a_4, -a_4, -a_4, a_4]$.

[0129] Each piece of data in the second group of eighth sequences $[b_1, b_2, b_3, b_4]$ is multiplied by the second group of first expanding sequences $[1, -1, 1, -1]$ and then spliced to obtain a second group of tenth sequences, i.e., $[b_1, -b_1, b_1, -b_1, b_2, -b_2, b_2, -b_2, b_3, -b_3, b_3, -b_3, b_4, -b_4, b_4, -b_4]$.

b3, b3, -b3, b4, -b4, b4, -b4].

[0130] Each piece of data in the third group of eighth sequences [c1, c2, c3, c4] is multiplied by the third group of first expanding sequences [1,1,-1,-1] and then spliced to obtain a third group of tenth sequences, i.e., [c1, c1, -c1, -c1, c2, c2, -c2, -c2, c3, c3, -c3, -c3, c4, c4, -c4, -c4].

[0131] Weight factors P(1), P(2), and P(3) are respectively configured for the above three groups of tenth sequences, and values of P(1), P(2), and P(3) may be flexibly set according to an application scenario, for example, P(1)=P(2)=P(3)=1 is selected. The three groups of tenth sequences are respectively multiplied by corresponding weight factors, and then added to obtain an eleventh sequence including 16 pieces of data, and the 16 pieces of data are spliced with rest 48 pieces of data of the second group of first sequences to obtain a second group of second sequences including 64 pieces of data.

[0132] For the third group of first sequences, the first expanding sequences with p=3 and q=4 are also selected, i.e., the three groups of first expanding sequences respectively are: [1,-1,-1,1], [1,-1,1,-1], [1,1,-1,-1].

[0133] For the third group of first sequences, the first 6 pieces of data are divided into three groups of eighth sequences, a first group of eighth sequences is [d1, d2], a second group of eighth sequences is [e1, e2], a third group of eighth sequences is [f1, f2].

[0134] Each piece of data in the first group of eighth sequences [d1, d2] is multiplied by the first group of first expanding sequences [1,-1,-1,1] and then spliced to obtain a first group of tenth sequences, i.e., [d1, -d1, -d1, d1, d2, -d2, -d2, d2].

[0135] Each piece of data in the second group of eighth sequences [e1, e2] is multiplied by the second group of first expanding sequences [1,-1,1,-1] and then spliced to obtain a second group of tenth sequences, i.e., [e1, -e1, e1, -e1, e2, -e2, e2, -e2].

[0136] Each piece of data in the third group of eighth sequences [f1, f2] is multiplied by the third group of first expanding sequences [1,1,-1,-1] and then spliced to obtain a third group of tenth sequences, i.e., [f1, f1, -f1, -f1, f2, f2, -f2, -f2].

[0137] Weight factors P(1), P(2), and P(3) are respectively configured for the above three groups of tenth sequences, and values of P(1), P(2), and P(3) may be flexibly set according to an application scenario, for example, P(1)=P(2)=P(3)=1 is selected. The three groups of tenth sequences are respectively multiplied by corresponding weight factors, and then added to obtain an eleventh sequence including 8 pieces of data, and the 8 pieces of data are spliced with rest 24 pieces of data of the third group of first sequences to obtain a third

group of second sequences including 32 pieces of data.

[0138] For outer edges of the first group and the fourth group of first sequences, orthogonal spreading code transformation may be or not be performed, and in the example, a case where orthogonal spreading code transformation being not performed on the outer edges of the first group and the fourth group of first sequences is selected, thus the first group of first sequences is the same as the first group of second sequences, the fourth group of first sequences is the same as the fourth group of second sequences.

[0139] First inverse Fourier transform with 2-fold oversampling is performed on the first group of second sequences to obtain a first group of third sequences, a total number of inverse Fourier transform points is 128, and a total number of pieces of data contained in the first group of third sequences is 128.

[0140] First inverse Fourier transform with 2-fold oversampling is performed on the second group of second sequences to obtain a second group of third sequences, a total number of inverse Fourier transform points is 128, a total number of pieces of data contained in the second group of third sequences is 128.

[0141] First inverse Fourier transform with 2-fold oversampling is performed on the third group of second sequences to obtain a third group of third sequences, a total number of inverse Fourier transform points is 64, a total number of pieces of data contained in the third group of third sequences is 64.

[0142] First inverse Fourier transform with 2-fold oversampling is performed on the fourth group of second sequences to obtain a fourth group of third sequences, a total number of inverse Fourier transform points is 64, a total number of pieces of data contained in the fourth group of third sequences is 64.

[0143] The first group of third sequences is used as a first group of fourth sequences, and the second group of third sequences is used as a second group of fourth sequences.

[0144] The third group of third sequences are spliced with another third group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with 2-fold oversampling on another first data sequence to be transmitted, to obtain a third group of fourth sequences.

[0145] The fourth group of third sequences are spliced with another fourth group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with 2-fold oversampling on another first data sequence to be transmitted, to obtain a fourth group of fourth sequences.

[0146] Second inverse Fourier transform is performed on the four groups of fourth sequences to obtain a second data sequence.

[0147] The second data sequence is transmitted on the N time-frequency resource blocks.

[0148] Example 2

5 [0149] In the example, a first data sequence to be transmitted is divided into four groups of first sequences, a first group of first sequences contains 64 pieces of data, a second group of first sequences contains 48 pieces of data, a third group of first sequences contains 24 pieces of data, a fourth group of first sequences contains 32 pieces of data, as shown in Fig. 3.

10 [0150] The data in the first group and the second group of first sequences are transmitted on the time-frequency resource blocks with a subcarrier spacing of 15kHz, and the data in the third group and the fourth group of first sequences are transmitted on the time-frequency resource blocks with a subcarrier spacing of 30 kHz.

[0151] In order to suppress interference between the time-frequency resource blocks with
15 different subcarrier spacings, orthogonal spreading code transformation is respectively performed on last 16 pieces of data in the data contained in the second group of first sequences and first 8 pieces of data in the data contained in the third group of first sequences, and a specific process is as follows.

[0152] First expanding sequences with $p=2$ and $q=4$ are selected, i.e., the two groups of
20 first expanding sequences respectively are: $[1, -1, -1, 1]$, $[1, -1, 1, -1]$.

[0153] For the second group of first sequences, the last 16 pieces of data are divided into two groups of eighth sequences, a first group of eighth sequences is $[a_1, a_2, a_3, \dots, a_8]$, a second group of eighth sequences is $[b_1, b_2, b_3, \dots, b_8]$.

[0154] Each piece of data in the first group of eighth sequences $[a_1, a_2, a_3, \dots, a_8]$ is
25 multiplied by the first group of first expanding sequences $[1, -1, -1, 1]$ and then spliced to obtain a first group of tenth sequences, i.e., $[a_1, -a_1, -a_1, a_1, a_2, -a_2, -a_2, a_2, a_3, -a_3, -a_3, a_3, \dots, a_8, -a_8, -a_8, a_8]$.

[0155] Each piece of data in the second group of eighth sequences $[b_1, b_2, b_3, \dots, b_8]$ is
multiplied by the second group of first expanding sequences $[1, -1, 1, -1]$ and then spliced
30 to obtain a second group of tenth sequences, i.e., $[b_1, -b_1, b_1, -b_1, b_2, -b_2, b_2, -b_2, b_3, -b_3, b_3, -b_3, \dots, b_8, -b_8, b_8, -b_8]$.

[0156] Weight factors $P(1)$ and $P(2)$ are respectively configured for the above two groups of tenth sequences, and values of $P(1)$ and $P(2)$ may be flexibly set according to an

application scenario, for example, $P(1)=P(2)=1$ is selected. The two groups of tenth sequences are respectively multiplied by corresponding weight factors, and then added to obtain an eleventh sequence including 32 pieces of data, and the 32 pieces of data are spliced with rest 32 pieces of data of the second group of first sequences to obtain a second group of second sequences including 64 pieces of data.

[0157] For the third group of first sequences, the first expanding sequences with $p=2$ and $q=4$ are also selected, i.e., the two groups of first expanding sequences respectively are: $[1,-1,-1,1]$, $[1,-1,1,-1]$.

[0158] For the third group of first sequences, the first 8 pieces of data are divided into two groups of eighth sequences, a first group of eighth sequences is $[c1, c2, c3, c4]$, a second group of eighth sequences is $[d1, d2, d3, d4]$.

[0159] Each piece of data in the first group of eighth sequences $[c1, c2, c3, c4]$ is multiplied by the first group of first expanding sequences $[1,-1,-1,1]$ and then spliced to obtain a first group of tenth sequences, i.e., $[c1, -c1, -c1, c1, c2, -c2, -c2, c2, c3, -c3, -c3, c3, c4, -c4, -c4, c4]$

[0160] Each piece of data in the second group of eighth sequences $[d1, d2, d3, d4]$ is multiplied by the second group of first expanding sequences $[1,-1,1,-1]$ and then spliced to obtain a second group of tenth sequences, i.e., $[d1, -d1, d1, -d1, d2, -d2, d2, -d2, d3, -d3, d3, -d3, d4, -d4, d4, -d4]$.

[0161] Weight factors $P(1)$ and $P(2)$ are respectively configured for the above two groups of tenth sequences, and values of $P(1)$ and $P(2)$ may be flexibly set according to an application scenario, for example, $P(1)=P(2)=1$ is selected. The two groups of tenth sequences are respectively multiplied by corresponding weight factors, and then added to obtain an eleventh sequence including 16 pieces of data, and the 16 pieces of data are spliced with rest 16 pieces of data of the third group of first sequences to obtain a third group of second sequences including 32 pieces of data.

[0162] In the example, the first group of first sequences is the same as a first group of second sequences, the fourth group of first sequences is the same as a fourth group of second sequences.

[0163] First inverse Fourier transform with 2-fold oversampling is performed on the first group of second sequences to obtain a first group of third sequences, a total number of inverse Fourier transform points is 128, and a total number of pieces of data contained in the first group of third sequences is 128.

[0164] First inverse Fourier transform with 2-fold oversampling is performed on the second group of second sequences to obtain a second group of third sequences, a total number of inverse Fourier transform points is 128, and a total number of pieces of data contained in the second group of third sequences is 128.

5 [0165] First inverse Fourier transform with 2-fold oversampling is performed on the third group of second sequences to obtain a third group of third sequences, a total number of inverse Fourier transform points is 64, and a total number of pieces of data contained in the third group of third sequences is 64.

10 [0166] First inverse Fourier transform with 2-fold oversampling is performed on the fourth group of second sequences to obtain a fourth group of third sequences, a total number of inverse Fourier transform points is 64, and a total number of pieces of data contained in the fourth group of third sequences is 64.

[0167] The first group of third sequences is used as a first group of fourth sequences, and the second group of third sequences is used as a second group of fourth sequences.

15 [0168] The third group of third sequences are spliced with another third group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with 2-fold oversampling on another first data sequence to be transmitted, to obtain a third group of fourth sequences.

20 [0169] The fourth group of third sequences are spliced with another fourth group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with 2-fold oversampling on another first data sequence to be transmitted, to obtain a fourth group of fourth sequences.

[0170] Second inverse Fourier transform is performed on the four groups of fourth sequences to obtain a second data sequence.

25 [0171] The second data sequence is transmitted on the N time-frequency resource blocks.

[0172] Example 3

30 [0173] In the example, a first data sequence to be transmitted is divided into four groups of first sequences, a first group of first sequences contains 60 pieces of data, a second group of first sequences contains 28 pieces of data, a third group of first sequences contains 56 pieces of data, a fourth group of first sequences contains 30 pieces of data, as shown in Fig. 4.

[0174] The data in the first group and the third group of first sequences are transmitted on the time-frequency resource blocks with a subcarrier spacing of 15kHz, and the data in

the second group and the fourth group of first sequences are transmitted on the time-frequency resource blocks with a subcarrier spacing of 30 kHz.

[0175] In order to suppress interference between the time-frequency resource blocks with different subcarrier spacings, orthogonal spreading code transformation is respectively performed on last 12 pieces of data in the data contained in the first group of first sequences, first 6 pieces of data and last 6 pieces of data in the data contained in the second group of first sequences, first 12 pieces of data and last 12 pieces of data in the data contained in the third group of first sequences, and first 6 pieces of data in the data contained in the fourth group of first sequences, and a specific process is as follows.

[0176] First expanding sequences with $p=3$ and $q=4$ are selected, i.e., the three groups of first expanding sequences respectively are: $[1, -1, -1, 1]$, $[1, -1, 1, -1]$, $[1, 1, -1, -1]$.

[0177] For the first group of first sequences, the last 12 pieces of data are divided into three groups of eighth sequences, a first group of eighth sequences is $[a_1, a_2, a_3, a_4]$, a second group of eighth sequences is $[b_1, b_2, b_3, b_4]$, a third group of eighth sequences is $[c_1, c_2, c_3, c_4]$.

[0178] Each piece of data in the first group of eighth sequences $[a_1, a_2, a_3, a_4]$ is multiplied by the first group of first expanding sequences $[1, -1, -1, 1]$ and then spliced to obtain a first group of tenth sequences, i.e., $[a_1, -a_1, -a_1, a_1, a_2, -a_2, -a_2, a_2, a_3, -a_3, -a_3, a_3, a_4, -a_4, -a_4, a_4]$.

[0179] Each piece of data in the second group of eighth sequences $[b_1, b_2, b_3, b_4]$ is multiplied by the second group of first expanding sequences $[1, -1, 1, -1]$ and then spliced to obtain a second group of tenth sequences, i.e., $[b_1, -b_1, b_1, -b_1, b_2, -b_2, b_2, -b_2, b_3, -b_3, b_3, -b_3, b_4, -b_4, b_4, -b_4]$.

[0180] Each piece of data in the third group of eighth sequences $[c_1, c_2, c_3, c_4]$ is multiplied by the third group of first expanding sequences $[1, 1, -1, -1]$ and then spliced to obtain a third group of tenth sequences, i.e., $[c_1, c_1, -c_1, -c_1, c_2, c_2, -c_2, -c_2, c_3, c_3, -c_3, -c_3, c_4, c_4, -c_4, -c_4]$.

[0181] Weight factors $P(1)$, $P(2)$, and $P(3)$ are respectively configured for the above three groups of tenth sequences, and values of $P(1)$, $P(2)$, and $P(3)$ may be flexibly set according to an application scenario, for example, $P(1)=P(2)=P(3)=1$ is selected. The three groups of tenth sequences are respectively multiplied by corresponding weight factors, and then added to obtain an eleventh sequence including 16 pieces of data, and the 16 pieces of data are spliced with rest 48 pieces of data of the first group of first sequences to obtain a

first group of second sequences including 64 pieces of data.

[0182] For the second group of first sequences, the first expanding sequences with $p=3$ and $q=4$ are also selected, i.e., the three groups of first expanding sequences respectively are: $[1,-1,-1,1]$, $[1,-1,1,-1]$, $[1,1,-1,-1]$.

5 [0183] For the second group of first sequences, the first 6 pieces of data are divided into three groups of eighth sequences, a first group of eighth sequences is $[d1, d2]$, a second group of eighth sequences is $[e1, e2]$, a third group of eighth sequences is $[f1, f2]$.

[0184] Each piece of data in the first group of eighth sequences $[d1, d2]$ is multiplied by the first group of first expanding sequences $[1,-1,-1,1]$ and then spliced to obtain a first
10 group of tenth sequences, i.e., $[d1, -d1, -d1, d1, d2, -d2, -d2, d2]$.

[0185] Each piece of data in the second group of eighth sequences $[e1, e2]$ is multiplied by the second group of first expanding sequences $[1,-1,1,-1]$ and then spliced to obtain a second group of tenth sequences, i.e., $[e1, -e1, e1, -e1, e2, -e2, e2, -e2]$.

[0186] Each piece of data in the third group of eighth sequences $[f1, f2]$ is multiplied by
15 the third group of first expanding sequences $[1,1,-1,-1]$ and then spliced to obtain a third group of tenth sequences, i.e., $[f1, f1, -f1, -f1, f2, f2, -f2, -f2]$.

[0187] Weight factors $P(1)$, $P(2)$, and $P(3)$ are respectively configured for the above three groups of tenth sequences, and values of $P(1)$, $P(2)$, and $P(3)$ may be flexibly set according to an application scenario, for example, $P(1)=P(2)=P(3)=1$ is selected. The three groups
20 of tenth sequences are respectively multiplied by corresponding weight factors, and then added to obtain an eleventh sequence including 8 pieces of data.

[0188] For the second group of first sequences, the last 6 pieces of data are divided into three groups of eighth sequences, a first group of eighth sequences is $[g1, g2]$, a second group of eighth sequences is $[h1, h2]$, a third group of eighth sequences is $[i1, i2]$.

25 [0189] Each piece of data in the first group of eighth sequences $[g1, g2]$ is multiplied by the first group of first expanding sequences $[1,-1,-1,1]$ and then spliced to obtain a first group of tenth sequences, i.e., $[g1, -g1, -g1, g1, g2, -g2, -g2, g2]$.

[0190] Each piece of data in the second group of eighth sequences $[h1, h2]$ is multiplied by the second group of first expanding sequences $[1,-1,1,-1]$ and then spliced to obtain a
30 second group of tenth sequences, i.e., $[h1, -h1, h1, -h1, h2, -h2, h2, -h2]$.

[0191] Each piece of data in the third group of eighth sequences $[i1, i2]$ is multiplied by the third group of first expanding sequences $[1,1,-1,-1]$ and then spliced to obtain a third group of tenth sequences, i.e., $[i1, i1, -i1, -i1, i2, i2, -i2, -i2]$.

[0192] Weight factors $P(1)$, $P(2)$, and $P(3)$ are respectively configured for the above three groups of tenth sequences, and values of $P(1)$, $P(2)$, and $P(3)$ may be flexibly set according to an application scenario, for example, $P(1)=P(2)=P(3)=1$ is selected. The three groups of tenth sequences are respectively multiplied by corresponding weight factors, and then
5 added to obtain an eleventh sequence including 8 pieces of data.

[0193] The 8 pieces of data obtained by performing orthogonal spreading code transformation on the first 6 pieces of data of the second group of first sequences are spliced with rest 16 pieces of data, and the 8 pieces of data obtained by performing orthogonal spreading code transformation on the last 6 pieces of data to obtain a second
10 group of second sequences including 32 pieces of data.

[0194] Similarly, orthogonal spreading code transformation is performed on the first 12 pieces of data of the third group of first sequences to obtain 16 pieces of data, orthogonal spreading code transformation is performed on the last 12 pieces of data of the third group of first sequences to obtain 16 pieces of data.

[0195] The 16 pieces of data obtained by performing orthogonal spreading code transformation on the first 12 pieces of data of the third group of first sequences are spliced with rest 32 pieces of data, and the 16 pieces of data obtained by performing orthogonal spreading code transformation on the last 12 pieces of data to obtain a third
15 group of second sequences including 64 pieces of data.

[0196] Similarly, orthogonal spreading code transformation is performed on the first 6
20 pieces of data of the fourth group of first sequences to obtain 8 pieces of data.

[0197] The 8 pieces of data obtained by performing orthogonal spreading code transformation on the first 6 pieces of data of the second group of first sequences are spliced with rest 24 pieces of data to obtain a fourth group of second sequences including
25 32 pieces of data.

[0198] First inverse Fourier transform with 2-fold oversampling is performed on the first group of second sequences to obtain a first group of third sequences, a total number of inverse Fourier transform points is 128, and a total number of pieces of data contained in the first group of third sequences is 128.

[0199] First inverse Fourier transform with 2-fold oversampling is performed on the
30 second group of second sequences to obtain a second group of third sequences, a total number of inverse Fourier transform points is 64, and a total number of pieces of data contained in the second group of third sequences is 64.

[0200] First inverse Fourier transform with 2-fold oversampling is performed on the third group of second sequences to obtain a third group of third sequences, a total number of inverse Fourier transform points is 128, and a total number of pieces of data contained in the third group of third sequences is 128.

5 [0201] First inverse Fourier transform with 2-fold oversampling is performed on the fourth group of second sequences to obtain a fourth group of third sequences, a total number of inverse Fourier transform points is 64, and a total number of pieces of data contained in the fourth group of third sequences is 64.

[0202] The first group of third sequences is used as a first group of fourth sequences.

10 [0203] The second group of third sequences are spliced with another second group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with 2-fold oversampling on another first data sequence to be transmitted, to obtain a second group of fourth sequences.

[0204] The third group of third sequences is used as a third group of fourth sequences.

15 [0205] The fourth group of third sequences are spliced with another third group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with 2-fold oversampling on another first data sequence to be transmitted, to obtain a fourth group of fourth sequences.

20 [0206] Second inverse Fourier transform is performed on the four groups of fourth sequences to obtain a second data sequence.

[0207] The second data sequence is transmitted on the N time-frequency resource blocks.

[0208] Example 4

25 [0209] In the example, a first data sequence to be transmitted is divided into four groups of first sequences, a first group of first sequences contains 60 pieces of data, a second group of first sequences contains 24 pieces of data, a third group of first sequences contains 48 pieces of data, a fourth group of first sequences contains 30 pieces of data, as shown in Fig. 5.

30 [0210] The data in the first group and the second group of first sequences are transmitted on the time-frequency resource blocks with a subcarrier spacing of 15kHz, and the data in the third group and the fourth group of first sequences are transmitted on the time-frequency resource blocks with a subcarrier spacing of 30 kHz.

[0211] In order to suppress interference between the time-frequency resource blocks with different subcarrier spacings, orthogonal spreading code transformation is respectively

performed on last 12 pieces of data in the data contained in the first group of first sequences, all the data contained in the second group of first sequences, all the data contained in the third group of first sequences, and first 6 pieces of data in the data contained in the fourth group of first sequences, and a specific process is as follows.

5 [0212] First expanding sequences with $p=3$ and $q=4$ is selected, i.e., the three groups of first expanding sequences respectively are: $[1,-1,-1,1]$, $[1,-1,1,-1]$, $[1,1,-1,-1]$.

[0213] Orthogonal spreading code transformation is performed on the last 12 pieces of data of the first group of first sequences to obtain 16 pieces of data, which are spliced with rest 48 pieces of data to obtain a first group of second sequences including 64 pieces
10 of data.

[0214] Orthogonal spreading code transformation is performed on 24 pieces of data of the second group of first sequences to obtain 32 pieces of data, so that a second group of second sequences includes 32 pieces of data.

[0215] Orthogonal spreading code transformation is performed on 48 pieces of data of the
15 third group of first sequences to obtain 64 pieces of data, so that a third group of second sequences includes 64 pieces of data.

[0216] Orthogonal spreading code transformation is performed on the first 6 pieces of data of the fourth group of first sequences to obtain 8 pieces of data, which are spliced with rest 24 pieces of data to obtain a second group of second sequences including 32
20 pieces of data.

[0217] First inverse Fourier transform with 2-fold oversampling is performed on the first group of second sequences to obtain a first group of third sequences, a total number of inverse Fourier transform points is 128, and a total number of pieces of data contained in the first group of third sequences is 128.

25 [0218] First inverse Fourier transform with 2-fold oversampling is performed on the second group of second sequences to obtain a second group of third sequences, a total number of inverse Fourier transform points is 64, and a total number of pieces of data contained in the second group of third sequences is 64.

[0219] First inverse Fourier transform with 2-fold oversampling is performed on the third
30 group of second sequences to obtain a third group of third sequences, a total number of inverse Fourier transform points is 128, and a total number of pieces of data contained in the third group of third sequences is 128.

[0220] First inverse Fourier transform with 2-fold oversampling is performed on the

fourth group of second sequences to obtain a fourth group of third sequences, a total number of inverse Fourier transform points is 64, and a total number of pieces of data contained in the fourth group of third sequences is 64.

[0221] The first group of third sequences is used as a first group of fourth sequences.

5 [0222] The second group of third sequences are spliced with another second group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with 2-fold oversampling on another first data sequence to be transmitted, to obtain a second group of fourth sequences.

[0223] The third group of third sequences is used as a third group of fourth sequences.

10 [0224] The fourth group of third sequences are spliced with another fourth group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with 2-fold oversampling on another first data sequence to be transmitted, to obtain a fourth group of fourth sequences.

[0225] Second inverse Fourier transform is performed on the four groups of fourth
15 sequences to obtain a second data sequence.

[0226] The second data sequence is transmitted on the N time-frequency resource blocks.

[0227] Example 5

[0228] In the example, a first data sequence to be transmitted is divided into four groups of first sequences, a first group of first sequences contains 64 pieces of data, a second
20 group of first sequences contains 60 pieces of data, a third group of first sequences contains 30 pieces of data, a fourth group of first sequences contains 32 pieces of data, as shown in Fig. 6.

[0229] The data in the first group and the second group of first sequences are transmitted on the time-frequency resource blocks with a subcarrier spacing of 15kHz, and the data
25 in the third group and the fourth group of first sequences are transmitted on the time-frequency resource blocks with a subcarrier spacing of 30 kHz.

[0230] In order to suppress interference between the time-frequency resource blocks with different subcarrier spacings, orthogonal spreading code transformation is respectively performed on last 12 pieces of data in the data contained in the second group of first
30 sequences and first 6 pieces of data in the data contained in the third group of first sequences, part of data in each group are pilot frequencies, and the pilot frequencies are not located at the positions of the data to be performed the orthogonal spreading code transformation thereon. A specific process is as follows.

[0231] First expanding sequences with $p=3$ and $q=4$ are selected, i.e., the three groups of first expanding sequences respectively are: $[1,-1,-1,1]$, $[1,-1,1,-1]$, $[1,1,-1,-1]$.

[0232] After the orthogonal spreading code transformation, a total number of pieces of data contained in a first group of second sequences is 64, a total number of pieces of data contained in a second group of second sequences is 64, a total number of pieces of data contained in a third group of second sequences is 32, and a total number of pieces of data contained in a fourth group of second sequences is 32.

[0233] First inverse Fourier transform with 2-fold oversampling is performed on the first group of second sequences to obtain a first group of third sequences, a total number of inverse Fourier transform points is 128, and a total number of pieces of data contained in the first group of third sequences is 128.

[0234] First inverse Fourier transform with 2-fold oversampling is performed on the second group of second sequences to obtain a second group of third sequences, a total number of inverse Fourier transform points is 128, and a total number of pieces of data contained in the second group of third sequences is 128.

[0235] First inverse Fourier transform with 2-fold oversampling is performed on the third group of second sequences to obtain a third group of third sequences, a total number of inverse Fourier transform points is 64, and a total number of pieces of data contained in the third group of third sequences is 64.

[0236] First inverse Fourier transform with 2-fold oversampling is performed on the fourth group of second sequences to obtain a fourth group of third sequences, a total number of inverse Fourier transform points is 64, and a total number of pieces of data contained in the fourth group of third sequences is 64.

[0237] The first group of third sequences is used as a first group of fourth sequences, and the second group of third sequences is used as a second group of fourth sequences.

[0238] The third group of third sequences are spliced with another third group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with 2-fold oversampling on another first data sequence to be transmitted, to obtain a third group of fourth sequences.

[0239] The fourth group of third sequences are spliced with another fourth group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with 2-fold oversampling on another first data sequence to be transmitted, to obtain a fourth group of fourth sequences.

[0240] Second inverse Fourier transform is performed on the four groups of fourth sequences to obtain a second data sequence.

[0241] The second data sequence is transmitted on the N time-frequency resource blocks.

[0242] Example 6

5 [0243] In the example, a first data sequence to be transmitted is divided into four groups of first sequences, a first group of first sequences contains 64 pieces of data, a second group of first sequences contains 60 pieces of data, a third group of first sequences contains 30 pieces of data, a fourth group of first sequences contains 32 pieces of data, as shown in Fig. 7.

10 [0244] The data in the first group and the second group of first sequences are transmitted on the time-frequency resource blocks with a subcarrier spacing of 15kHz, and the data in the third group and the fourth group of first sequences are transmitted on the time-frequency resource blocks with a subcarrier spacing of 30 kHz.

[0245] In order to suppress interference between the time-frequency resource blocks with
15 different subcarrier spacings, orthogonal spreading code transformation is respectively performed on last 12 pieces of data in the data contained in the second group of first sequences and first 6 pieces of data in the data contained in the third group of first sequences, and a specific process is as follows.

[0246] First expanding sequences with $p=3$ and $q=4$ are selected, i.e., the three groups of
20 first expanding sequences respectively are: $[1,-1,-1,1]$, $[1,-1,1,-1]$, $[1,1,-1,-1]$.

[0247] After the orthogonal spreading code transformation, a total number of pieces of data contained in a first group of second sequences is 64, a total number of pieces of data contained in a second group of second sequences is 64, a total number of pieces of data contained in a third group of second sequences is 32, a total number of pieces of data
25 contained in a fourth group of second sequences is 32.

[0248] First inverse Fourier transform with 2-fold oversampling is performed on the first group of second sequences to obtain a first group of third sequences, a total number of inverse Fourier transform points is 128, and a total number of pieces of data contained in the first group of third sequences is 128.

30 [0249] First inverse Fourier transform with 2-fold oversampling is performed on the second group of second sequences to obtain a second group of third sequences, a total number of inverse Fourier transform points is 128, and a total number of pieces of data contained in the second group of third sequences is 128.

[0250] First inverse Fourier transform with 2-fold oversampling is performed on the third group of second sequences to obtain a third group of third sequences, a total number of inverse Fourier transform points is 64, and a total number of pieces of data contained in the third group of third sequences is 64.

5 [0251] First inverse Fourier transform with 2-fold oversampling is performed on the fourth group of second sequences to obtain a fourth group of third sequences, a total number of inverse Fourier transform points is 64, and a total number of pieces of data contained in the fourth group of third sequences is 64.

[0252] The first group of third sequences is used as a first group of fourth sequences, and
10 the second group of third sequences is used as a second group of fourth sequences.

[0253] The third group of third sequences are spliced with another third group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with 2-fold oversampling on another first data sequence to be transmitted, to obtain a third group of fourth sequences.

15 [0254] The fourth group of third sequences are spliced with another fourth group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with 2-fold oversampling on another first data sequence to be transmitted, to obtain a fourth group of fourth sequences.

[0255] Second inverse Fourier transform is performed on the four groups of fourth
20 sequences and other data to obtain a second data sequence, the other data refer to data sequences obtained by performing the first inverse Fourier transform with 2-fold oversampling on the data sequences except the first data sequence.

[0256] The second data sequence is transmitted on the N time-frequency resource blocks.

[0257] Example 7

25 [0258] In the example, a first data sequence to be transmitted is divided into four groups of first sequences, a first group of first sequences contains 64 pieces of data, a second group of first sequences contains 60 pieces of data, a third group of first sequences contains 30 pieces of data, a fourth group of first sequences contains 32 pieces of data, as shown in Fig. 8.

30 [0259] The data in the first group and the second group of first sequences are transmitted on the time-frequency resource blocks with a subcarrier spacing of 15kHz, and the data in the third group and the fourth group of first sequences are transmitted on the time-frequency resource blocks with a subcarrier spacing of 30 kHz.

[0260] In order to suppress interference between the time-frequency resource blocks with different subcarrier spacings, orthogonal spreading code transformation is respectively performed on last 12 pieces of data in the data contained in the second group of first sequences and first 6 pieces of data in the data contained in the third group of first sequences, and a specific process is as follows.

[0261] First expanding sequences with $p=3$ and $q=4$ are selected, i.e., the three groups of first expanding sequences respectively are: $[1,-1,-1,1]$, $[1,-1,1,-1]$, $[1,1,-1,-1]$.

[0262] After the orthogonal spreading code transformation, a total number of pieces of data contained in a first group of second sequences is 64, a total number of pieces of data contained in a second group of second sequences is 64, a total number of pieces of data contained in a third group of second sequences is 32, a total number of pieces of data contained in a fourth group of second sequences is 32.

[0263] First inverse Fourier transform with 2-fold oversampling is performed on the first group of second sequences to obtain a first group of third sequences, a total number of inverse Fourier transform points is 128, and a total number of pieces of data contained in the first group of third sequences is 128.

[0264] First inverse Fourier transform with 2-fold oversampling is performed on the second group of second sequences to obtain a second group of third sequences, a total number of inverse Fourier transform points is 128, and a total number of pieces of data contained in the second group of third sequences is 128.

[0265] First inverse Fourier transform with 2-fold oversampling is performed on the third group of second sequences to obtain a third group of third sequences, a total number of inverse Fourier transform points is 64, and a total number of pieces of data contained in the third group of third sequences is 64.

[0266] First inverse Fourier transform with 2-fold oversampling is performed on the fourth group of second sequences to obtain a fourth group of third sequences, a total number of inverse Fourier transform points is 64, and a total number of pieces of data contained in the fourth group of third sequences is 64.

[0267] The first group of third sequences is used as a first group of fourth sequences, and the second group of third sequences is used as a second group of fourth sequences.

[0268] The third group of third sequences are spliced with another third group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with 2-fold oversampling on another first data sequence to be

transmitted, to obtain a third group of fourth sequences.

[0269] The fourth group of third sequences are spliced with another fourth group of third sequences, obtained by performing orthogonal spreading code transformation and first inverse Fourier transform with 2-fold oversampling on another first data sequence to be transmitted, to obtain a fourth group of fourth sequences.

[0270] Second inverse Fourier transform is performed on the four groups of fourth sequences to obtain a second data sequence, specifically, the four groups of fourth sequences are arranged in four rows respectively, data in a second row and a fourth row are respectively multiplied with coefficient $[1, -1, 1, \dots, (-1)^{128}]$; then taking four pieces of data out by columns, and inverse Fourier transform with 16-point oversampling is performed on the four pieces of data to obtain one sub-symbol, that is, every four pieces of data taken out by columns, after being subjected to the inverse Fourier transform with oversampling, corresponds to one sub-symbol, as shown in Fig. 8, the four pieces of data taken out at a first time corresponds to a sub-symbol 1, the four pieces of data taken out at a second time corresponds to a sub-symbol 2,, and the four pieces of data taken out at a 128th time corresponds to a sub-symbol 128. Each sub-symbol is then repetitively expanded by 4 times, as shown in Fig. 8, the sub-symbol 1 is expanded into 4 sub-symbols 1 to obtain a first group of sixth sequences, the sub-symbol 2 is expanded into 4 sub-symbols 2 to obtain a second group of sixth sequences,, the sub-symbol 128 is expanded into 4 sub-symbols 128 to obtain a 128th group of sixth sequences.

[0271] The 128 groups of sixth sequences are arranged in a misaligned manner into 128 rows, a misaligned interval between first pieces of data in two adjacent rows is half of a length of a sub-symbol, and data in a same column of the 128 rows are added up to obtain a second data sequence.

[0272] The second data sequence is transmitted on the N time-frequency resource blocks.

[0273] Example 8

[0274] In the example, after obtaining the second data sequence according to the above examples, filtering is performed on the second data sequence, and then Digital Analog Conversion (DAC) and radio frequency processing are performed on filtered second data sequence, and the second data sequence subjected to the DAC and radio frequency processing is transmitted on N time-frequency resource blocks.

[0275] The filtering is single-phase filtering or polyphase filtering, the polyphase filtering is to filter each group of new N groups of data sequences.

[0276] In a second aspect, referring to Fig. 9, an embodiment of the present application provides an electronic device, including: at least one processor 901; a memory 902 having at least one program stored thereon, the at least one program, executed by the at least one processor 901, implements the data transmission method described above.

5 [0277] In some implementations, the electronic device further includes: at least one I/O interface 903 connected between the processor 901 and the memory 902, and configured to implement information interaction between the processor 901 and the memory 902.

[0278] The processor 901 is a device having a capability of processing data, includes, but is not limited to, a Central Processing Unit (CPU), and the like; the memory 902 is a
10 device having a capability of storing data, includes, but is not limited to, a random access memory (RAM, in particular, SDRAM, DDR, and the like), a read only memory (ROM), an electrically erasable programmable read only memory (EEPROM), and a FLASH; the I/O interface (read/write interface) 903 is connected between the processor 101 and the memory 902, enables to implement information interaction between the processor 901 and
15 the memory 902, and includes, but is not limited to, a data Bus 904, and the like.

[0279] In some implementations, the processor 901, the memory 902, and the I/O interface 903 are connected together through a bus 904, and are then connected to other components of a computing device.

[0280] In a third aspect, an embodiment of the present application provides a computer
20 readable medium having a computer program stored thereon, the computer program, executed by a processor, implements the data transmission method described above.

[0281] It should be understood by those of ordinary skilled in the art that all or some of the operations in the method, , or the functional modules/components in the system or the device described above may be implemented as software, firmware, hardware, or suitable
25 combinations thereof; for example, one physical component may have a plurality of functions, or one function or operation may be performed through a cooperation of several physical components. Some or all of the physical components may be implemented as software executed by a processor, such as a central processing unit, a digital signal processor or a microprocessor, or may be implemented as hardware, or may be
30 implemented as an integrated circuit, such as an application specific integrated circuit. Such software may be distributed on a computer-readable medium, the computer-readable medium may include computer storage medium (or non-transitory medium) and communication medium (or transitory medium). As known to those skilled in the art, the

term of the computer storage medium includes volatile/nonvolatile or removable/non-removable medium used in any method or technology for storing information (such as computer-readable instructions, data structures, program modules and other data). The computer storage medium includes, but is not limited to, RAM, ROM, EEPROM, a flash
5 memory or other memory techniques, CD-ROM, a Digital Video Disk (DVD) or other optical discs, magnetic cassettes, magnetic tapes, magnetic disks or other magnetic storage devices, or any other medium which can be used to store the desired information and can be accessed by a computer. In addition, as known to those skilled in the art, the communication medium generally includes computer-readable instructions, data
10 structures, program modules or other data in a modulated data signal, such as a carrier wave or other transmission mechanism, and may include any information delivery medium.

[0282] The present application discloses the exemplary embodiments, and although specific terms are employed, they are used and should only be interpreted in a generic and
15 descriptive meaning and not for purposes of a limitation. In some examples, it is apparent to those skilled in the art that features, characteristics and/or elements described in connection with specific embodiments may be used alone or in combination with features, characteristics and/or elements described in connection with other embodiments, unless explicitly stated otherwise. Therefore, it should be understood by those skilled in the art
20 that various changes in form and details may be made without departing from the scope of the present application as set forth in the appended claims.

What is claimed is:

1. A data transmission method, comprising:

5 performing orthogonal spreading code transformation on data contained in first sequences satisfying a condition, in N groups of first sequences obtained by dividing a first data sequence to be transmitted, to generate N groups of second sequences, wherein, the N groups of first sequences correspond to N time-frequency resource blocks, and the first sequences satisfying the condition comprise first sequences, in the N groups of first sequences, corresponding to the time-frequency resource blocks with
10 a subcarrier spacing different from a subcarrier spacing of adjacent time-frequency resource blocks; N is an integer greater than or equal to 2;

performing first inverse Fourier transform with A-fold oversampling on the N groups of second sequences to obtain N groups of third sequences, wherein A is an integer greater than or equal to 1;

15 performing second inverse Fourier transform on the N groups of third sequences to obtain a second data sequence;

transmitting the second data sequence on the N time-frequency resource blocks.

2. The method according to claim 1, wherein after performing first inverse
20 Fourier transform with A-fold oversampling on the N groups of second sequences to obtain N groups of third sequences, and before performing second inverse Fourier transform on the N groups of third sequences to obtain the second data sequence, the method further comprises:

expanding the N groups of third sequences into N groups of fourth sequences,
25 wherein, a total number of pieces of data contained in each group of fourth sequences is B, and B is a maximum value of a total number of pieces of data contained in the N groups of third sequences,

the performing second inverse Fourier transform on the N groups of third sequences to obtain a second data sequence comprises:

30 performing second inverse Fourier transform on the N groups of fourth sequences to obtain the second data sequence.

3. The method according to claim 2, wherein the performing second inverse Fourier transform on the N groups of fourth sequences to obtain the second data sequence comprises:

performing second inverse Fourier transform with C-fold oversampling on N pieces of data consisting of a j^{th} data of the N groups of fourth sequences to obtain a j^{th} group of fifth sequences, wherein C is an integer greater than or equal to 1, j is an integer greater than or equal to 1 and less than or equal to B;

expanding B groups of fifth sequences into B groups of sixth sequences; wherein a total number of pieces of data contained in the sixth sequences is D times a total number of pieces of data contained in the fifth sequences, wherein D is an integer greater than or equal to 2;

splicing the B groups of sixth sequences into the second data sequence.

4. The method according to claim 3, wherein after expanding the B groups of fifth sequences into the B groups of sixth sequences, and before splicing the B groups of sixth sequences into the second data sequence, the method further comprises:

performing windowing operation on the B groups of sixth sequences,

the splicing the B groups of sixth sequences into the second data sequence comprises:

splicing the B groups of sixth sequences subjected to the windowing operation into the second data sequence.

5. The method according to claim 1, wherein after performing first inverse Fourier transform with A-fold oversampling on the N groups of second sequences to obtain N groups of third sequences, and before performing second inverse Fourier transform on the N groups of third sequences to obtain the second data sequence, the method further comprises:

performing coefficient correction on the N groups of third sequences to obtain N groups of seventh sequences,

the performing second inverse Fourier transform on the N groups of third sequences to obtain a second data sequence comprises:

performing second inverse Fourier transform on the N groups of seventh sequences.

6. The method according to claim 5, wherein the performing coefficient correction on the N groups of third sequences to obtain N groups of seventh sequences comprises:

5 multiplying even-numbered groups of third sequences in the N groups of third sequences by corresponding data of a first coefficient vector to obtain the N groups of seventh sequences; or

10 multiplying odd-numbered groups of third sequences in the N groups of third sequences by corresponding data of a second coefficient vector to obtain the N groups of seventh sequences.

7. The method according to claim 1, wherein after performing second inverse Fourier transform on the N groups of third sequences to obtain the second data sequence, and before transmitting the second data sequence on the N time-frequency resource blocks, the method further comprises:

15 filtering the second data sequence,
the transmitting the second data sequence on the N time-frequency resource blocks comprises:

20 transmitting the second data sequence filtered on the N time-frequency resource blocks.

8. The method according to any one of claims 1 to 7, wherein the performing orthogonal spreading code transformation on data contained in first sequences satisfying a condition, in N groups of first sequences obtained by dividing a first data sequence to be transmitted, to generate N groups of second sequences comprises:

25 performing orthogonal spreading code transformation on data to be transformed in the data contained in the first sequences satisfying the condition to obtain a corresponding second sequence, wherein, the data to be transformed is fore part data, or rear part data, or both the fore part data and the rear part data, or all data, in the data contained in the first sequences satisfying the condition.

30 9. The method according to claim 8, wherein the data to be transformed excludes reference signal data.

10. The method according to claim 8, wherein the data to be transformed comprises h pieces of data, and data obtained by performing orthogonal spreading code transformation on the data to be transformed comprises hq/p pieces of data;

5 wherein h is an integer greater than or equal to 1, p is a total number of groups of first expanding sequences used for the orthogonal spreading code transformation, and q is a length of the first expanding sequence.

10 11. The method according to claim 8, wherein the performing orthogonal spreading code transformation on data to be transformed in the data contained in the first sequences satisfying the condition to obtain a corresponding second sequence comprises:

15 expanding p groups of eighth sequences obtained by dividing the data to be transformed into p groups of ninth sequences with a length of hq/p , wherein h is an integer greater than or equal to 1, p is a total number of groups of first expanding sequences used for the orthogonal spreading code transformation, and q is a length of the first expanding sequences;

expanding p groups of first expanding sequences with a length of q into p groups of second expanding sequences with a length of hq/p ;

20 multiplying corresponding data in a k^{th} group of ninth sequences and a k^{th} group of second expanding sequences to obtain a k^{th} group of tenth sequences, wherein k is an integer greater than or equal to 1, and less than or equal to p ;

performing a weighted summation on p groups of tenth sequences to obtain an eleventh sequence;

25 splicing the eleventh sequence with other data except the data to be transformed in the data contained in the first sequences satisfying the condition to obtain the corresponding second sequence.

30 12. The method according to claim 8, wherein the performing orthogonal spreading code transformation on data to be transformed in the data contained in the first sequences satisfying the condition to obtain a corresponding second sequence comprises:

multiplying each piece of data in a k^{th} group of eighth sequences in p groups of eighth sequences, obtained by dividing the data to be transformed, by each piece of

data in a k^{th} group of first expanding sequences in p groups of first expanding sequences with a length of q to obtain a k^{th} group of tenth sequences, wherein p is a total number of groups of first expanding sequences used for the orthogonal spreading code transformation, and q is a length of the first expanding sequence, and k is an integer greater than or equal to 1, and less than or equal to p ;

performing a weighted summation on p groups of tenth sequences to obtain an eleventh sequence;

splicing the eleventh sequence with other data except the data to be transformed in the data contained in the first sequences satisfying the condition to obtain the corresponding second sequence.

13. The method according to any one of claims 1 to 7, wherein A is 2, the performing first inverse Fourier transform with A -fold oversampling on the N groups of second sequences to obtain N groups of third sequences comprises:

adding data of $\frac{k'(n)}{2}$ zeros at two ends of an n^{th} group of second sequences respectively, and performing first inverse Fourier transform on the n^{th} group of second sequences added with the data of zeros to obtain an n^{th} group of third sequences, wherein n is an integer greater than or equal to 1 and less than or equal to N , and $k'(n)$ is a total number of pieces of data contained in the n^{th} group of second sequences.

14. The method according to any one of claims 1 to 7, wherein the first sequences satisfying the condition further comprise: a first group of first sequences and an N^{th} group of first sequences.

15. The method according to any one of claims 1 to 7, wherein a subcarrier spacing of the time-frequency resource block corresponding to the first sequences satisfying the condition is different from a subcarrier spacing of a previous time-frequency resource block; or

the subcarrier spacing of the time-frequency resource block corresponding to the first sequences satisfying the condition is different from a subcarrier spacing of a next time-frequency resource block; or

the subcarrier spacing of the time-frequency resource block corresponding to

the first sequences satisfying the condition is different from subcarrier spacings of the previous time-frequency resource block and the next time-frequency resource block.

16. The method according to any one of claims 1 to 7, wherein a ratio of a total number of pieces of data contained in the first sequences satisfying the condition to a total number of pieces of data contained in a previous group of first sequences is 2 to the power of i , wherein i is an integer greater than or equal to 1; or

a ratio of the number of pieces of data contained in the first sequences satisfying the condition to a total number of pieces of data contained in a next group of first sequences is 2 to the power of i ; or

the ratio of the number of pieces of data contained in the first sequences satisfying the condition to the number of pieces of data contained in the previous group of first sequences is 2 to the power of i , and the ratio of the number of pieces of data contained in the first sequences satisfying the condition to the number of pieces of data contained in the next group of first sequences is 2 to the power of i .

17. The method according to any one of claims 1 to 7, wherein a total number of pieces of data contained in each group of second sequences is 2 to the power of i , wherein i is an integer greater than or equal to 1.

18. The method according to any one of claims 1 to 7, wherein a ratio of numbers of pieces of data contained in any two groups of second sequences is 2 to the power of i , wherein i is an integer greater than or equal to 1.

19. The method according to any one of claims 1 to 7, wherein there are at least two groups of second sequences containing different numbers of pieces of data.

20. The method according to any one of claims 1 to 7, wherein a total number of inverse Fourier transform points for the first inverse Fourier transform performed on an n^{th} group of second sequences is the same as a total number of pieces of data contained in an n^{th} group of third sequences, wherein n is an integer greater than or equal to 1 and less than or equal to N .

21. The method according to any one of claims 1 to 7, wherein a ratio of numbers of inverse Fourier transform points for the first inverse Fourier transform performed on any two groups of second sequences is 2 to the power of i , wherein i is an integer greater than or equal to 1.

5

22. The method according to any one of claims 1 to 7, wherein there are at least two groups of second sequences with different numbers of inverse Fourier transform points for the first inverse Fourier transform performed thereon.

10

23. The method according to any one of claims 1 to 7, wherein a zero-frequency position of the first inverse Fourier transform performed on an n^{th} group of second sequences is within a frequency range of the time-frequency resource block corresponding to an n^{th} group of first sequences, wherein n is an integer greater than or equal to 1 and less than or equal to N ;

15

zero-frequency positions of the first inverse Fourier transform performed on different groups of second sequences are different.

24. The method according to any one of claims 1 to 7, wherein a zero-frequency position of the first inverse Fourier transform performed on an n^{th} group of second sequences is in one of subcarriers contained in the time-frequency resource block corresponding to an n^{th} group of first sequences, wherein n is an integer greater than or equal to 1 and less than or equal to N .

20

25. The method according to any one of claims 1 to 7, wherein a total number of pieces of data contained in the third sequence corresponding to the first sequences satisfying the condition is greater than twice a total number of subcarriers contained in the time-frequency resource block corresponding to the first sequences satisfying the condition, a total number of pieces of data contained in the third sequence corresponding to first sequences not satisfying the condition is twice a total number of subcarriers contained in the time-frequency resource block corresponding to the first sequences not satisfying the condition.

25

30

26. The method according to any one of claims 1 to 7, wherein a total number of inverse Fourier transform points for the second inverse Fourier transform performed on N groups of third sequences is greater than N.

5 27. The method according to any one of claims 1 to 7, wherein the N time-frequency resource blocks are part or all of time-frequency resource blocks in a channel bandwidth, and the first data sequence to be transmitted is part or all of data to be transmitted in the channel bandwidth.

10 28. An electronic device, comprising:
at least one processor;
a memory having at least one program stored thereon, the at least one program, executed by the at least one processor, implements the data transmission method of any one of claims 1 to 27.

15 29. A computer readable medium having a computer program stored thereon, the computer program, executed by a processor, implements the data transmission method of any one of claims 1 to 27.