

SCIENTIFIC REPORTS



OPEN

***HBP Builder*: A Tool to Generate Hyperbranched Polymers and Hyperbranched Multi-Arm Copolymers for Coarse-grained and Fully Atomistic Molecular Simulations**

Received: 14 March 2016

Accepted: 28 April 2016

Published: 18 May 2016

Chunyang Yu, Li Ma, Shanlong Li, Haina Tan, Yongfeng Zhou & Deyue Yan

Computer simulation has been becoming a versatile tool that can investigate detailed information from the microscopic scale to the mesoscopic scale. However, the crucial first step of molecular simulation is model building, particularly for hyperbranched polymers (HBPs) and hyperbranched multi-arm copolymers (HBMCs) with complex and various topological structures. Unlike well-defined polymers, not only the molar weight of HBPs/HBMCs with polydispersity, but the HBPs/HBMCs with the same degree of polymerization (*DP*) and degree of branching (*DB*) also have many possible topological structures, thus making difficulties for user to build model in molecular simulation. In order to build a bridge between model building and molecular simulation of HBPs and HBMCs, we developed *HBP Builder*, a C language open source HBPs/HBMCs building toolkit. *HBP Builder* implements an automated protocol to build various coarse-grained and fully atomistic structures of HBPs/HBMCs according to user's specific requirements. Meanwhile, coarse-grained and fully atomistic output structures can be directly employed in popular simulation packages, including *HOOMD*, *Tinker* and *Gromacs*. Moreover, *HBP Builder* has an easy-to-use graphical user interface and the modular architecture, making it easy to extend and reuse it as a part of other program.

Hyperbranched polymers (HBPs) as a polymer with a three-dimensional and highly branched globular structure have emerged during the past twenty years^{1–4}. Unlike well-defined dendrimers, HBPs have linear units in structures except dendritic and terminal units, and the degree of branching (*DB*) of them are usually less than 0.5. So, HBPs are not as regular as dendrimers. However, HBPs also have the similar properties to dendrimers when compared with the linear counterparts, including a large population of functional groups, lower solution or melt viscosity, and better solubility^{4,5}. In addition, the synthesis of HBPs generally involves one-pot reaction without the troublesome and time-consuming separation and purification process like dendrimers^{6–11}, which has greatly reduced the cost. Thus, HBPs have attracted great interest from both the scientific and industrial communities, and have found many applications in various fields such as coatings^{12,13}, additives^{14,15}, drug delivery¹⁶, supramolecular chemistry^{17,18}, and so on. In recent year, amphiphilic hyperbranched multi-arm copolymers (HBMCs) (see Figure S1 in supplementary information) consisting of a hyperbranched core and many linear arms have been proved to be excellent self-assembly precursors, and many delicate supramolecular structures with all scales and dimensions have been prepared through the solution self-assembly of amphiphilic HBMCs^{16,19–22}.

Although the rapid progress has been made in experimental studies of HBPs and HBMCs, there are still many problems such as the static and dynamic properties of HBPs/HBMCs at micro level have not been fully understood due to the limitation of experimental means. For example, the conformational behavior of individual HBPs/

School of Chemistry and Chemical Engineering, State Key Laboratory of Metal Matrix Composites, Shanghai Jiao Tong University, 800 Dongchuan Road, Shanghai 200240, P. R. China. Correspondence and requests for materials should be addressed to Y.Z. (email: yfzhou@sjtu.edu.cn)

HBMCs in solution, the interaction mechanism between HBPs/HBMCs and small molecules or polymers (especially the biopolymers like DNAs or RNAs), and the conformation and phase separation behaviors of individual HBMCs during the self-assembly processes *et al.*^{16,21,22}. Thus, it is necessary to resort to the theoretic calculations or simulations to fully study HBPs and HBMCs.

In fact, nowadays, computer simulation has been becoming a powerful tool to investigate molecular information at any required level and has been a necessary complementary to experimental method. Different simulation methods were developed for different length and time scales. *Ab initio* molecular dynamics were developed to study complex processes at the electronic level, including the breaking and making of chemical bonds, by means of purely electronic structure theory^{23,24}. The classical all-atom (AA) simulation methods^{25–30}, which describe the atomic-level detail, were used to simulate the conformation and dynamical behaviors of polymers. Due to the current computational power, AA simulations are often constrained in time scale up to 100 ns and limited in the number of atoms. Thus the coarse-grained (CG) methods have been developed to simulate the aggregate behaviors of large number of molecules in bulk or in solution^{31–34}. However, polymers are complex macromolecules whose spatial scales can vary from the atomistic level to the scale of tens of nanometers. The corresponding time scales of the dynamic processes relevant for different materials properties span an even wider range, from femtoseconds to milliseconds, seconds or for large scale ordering processes such as phase separation in blends. The spatial and temporal scales become even larger when the polymers form aggregates during the self-assembly process. Thus, no single model or simulation algorithm currently available can encompass this range of length and time scales. To address this issue, some multiscale molecular modeling methods have been developed and applied to simulate polymers^{35–39}.

Although significant progresses have been made for simulation methods, considering the special molecular structure of HBPs/HBMCs, the simulation studies of them are still quite limited. In order to perform the computer simulations on HBPs/HBMCs, the first necessary step is model building, which is quite difficult. A famous dendrimer builder toolkit (DBT) for generating dendrimer configuration of desired generation and with various architectures was developed by Vishal Maingi and co-workers⁴⁰. However, unlike these monodisperse polymers, HBPs are a mixture of thousands of different topological polymers. In experiments, for both HBP solution and bulk, it is a mixture of many HBP molecules with a polydispersity of molecular weight, a polydispersity of DB, and a distribution of topological structure. Furthermore, it is even more difficult to describe HBMCs, which should not only contain the three structure parameters as mentioned above for conventional HBPs, but also should include the distributions of the length of the arms (polydispersity of molecular weight), the number of the arms (graft ratio), and the graft position. It is very challengeable to set up a general toolbox to fully describe HBPs and HBMCs.

So far, due to the very complex topologic structures, most of the reported theoretic models on HBPs/HBMCs are based on coarse-grained method. In summary, up to now, there are three types of theoretic models. The simplest type of them is to regard HBPs/HBMCs as the monodisperse polymers. For example, in the dissipative particle dynamics (DPD) simulations on the self-assembly of amphiphilic HBMCs, all the polymers have the same predetermined hyperbranched structure in the simulation box^{41,42}. This monodisperse model is quite simple but is far from the fact. The second type of the models involves building HBPs randomly and then selecting several representative polymers for further study. The sequential addition method is the representative method of this type. This method was first developed for building HBPs to simulate its intrinsic viscosity with varying topologies by Davies *et al.*⁴³. Since then, this method was widely used to investigate the conformational behavior of pure HBPs and HBP complexes. For example, HBPs under shear flow^{44,45} and elongational flow⁴⁶, the conformational behavior of charged/uncharged HBPs in solution⁴⁷ and the influence of the Wiener index⁴⁸ on the intrinsic viscosity and radius of gyration⁴⁹ were investigated by Adolf and Karatasos *et al.* Nevertheless, these models are still deviated from the actual polydisperse HBPs. The third type of models has sincerely considered the polydispersity of HBPs, and was developed by Ricardo Rodríguez Schmidt and coworkers^{50,51}. In this model, they firstly obtained the coarse-grained model parameters from atomic-level simulations of small chains fragments, and then they used the Monte Carlo technique to generate a set of polydisperse HBPs with required DBs. This multi-scale simulation method has the advantage of investigating conformational and dynamic behavior of HBPs by a very simple coarse-grained model and none of the parameters of that model need to adjust to fit experimental data.

Although the abovementioned three coarse graining methods are efficient and can allow the exploration of the behavior of HBPs/HBMCs over longer times, the loss of the atomistic information is inevitable. However, in many cases atomistic insight is required when special interactions need to be considered, for example, the non-covalent interactions like hydrogen bonds or specific chemical interactions among polymers when the HBPs or HBMCs are in the self-assembly process^{17,18}. In addition, in the biomedical applications of HBPs/HBMCs or their self-assemblies, the interactions between them and drugs, DNAs or proteins should have an atomistic resolution¹⁶. Moreover, electrostatic interaction play a very important role in the budding and fission process of HBPs/HBMCs vesicle^{52,53}, the atomistic resolution are essential for recognizing the microscopic regulating mechanism. Therefore, it is very useful to combine coarse-grained and atomistic methods together in the computer simulations on HBPs/HBMCs.

Based on the above experimental and theoretical studies, there are two crucial problems need to be considered for developing a precise model for HBPs and HBMCs. On the one hand, the polydispersity of HBPs and HBMCs should be fully taken into account, including the polydispersity of molecular weight, DB, and distribution of topological structure; on the other hand, developing computational model for HBPs and HBMCs should combine coarse-grained and all-atom models together, especially in the case where both the aggregated and unimolecular information are important. However, to our knowledge, up to now no theoretic toolbox with all these functions has been reported for HBPs/HBMCs.

To address this issue, we developed *HBP Builder*, a C language open source HBP/HBMC building toolkit. *HBP Builder* implements an automated protocol to build coarse-grained and fully atomistic structures of HBPs/

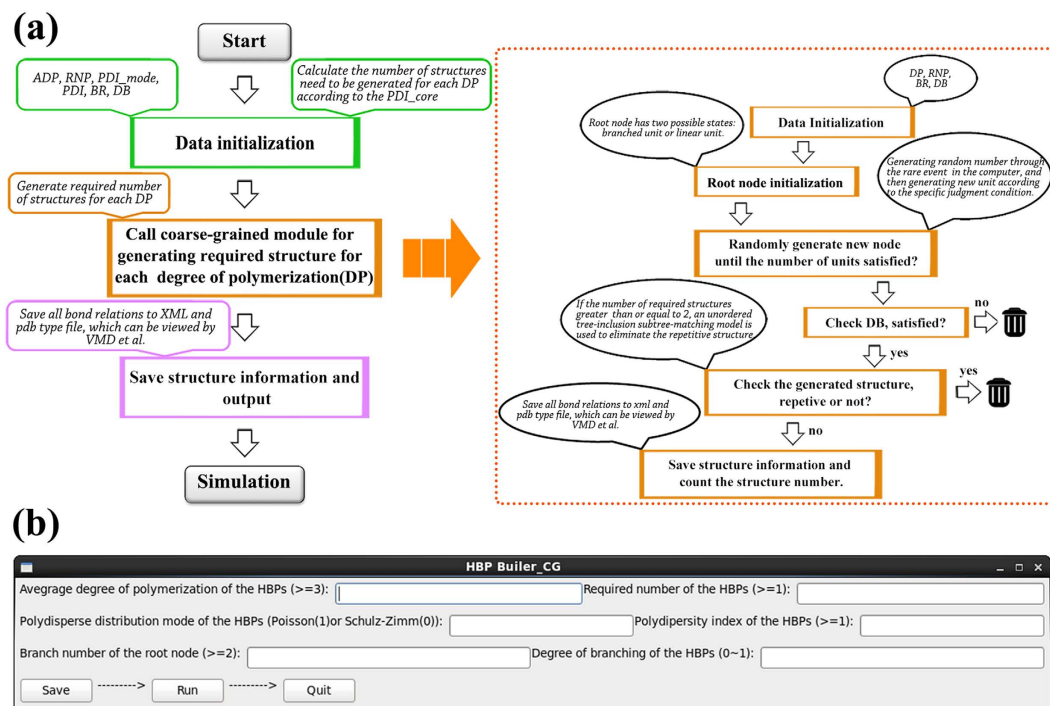


Figure 1. The flow chart of HBP Builder_CG module for generating HBPs. (a) The interactive input graphic interface of HBP Builder_CG module (b).

HBMCs according to user's specific requirements. The HBP Builder program mainly includes two function modules: one of them is HBP Builder_CG, which is developed for quickly generating thousands of polydisperse coarse-grained HBPs and HBMCs with required average PDI, DB and degree of polymerization (DP); the other is HBP Builder_AA, which is developed for converting coarse-grained HBPs and HBMCs to fully atomistic structures. In order to improve operation, we designed an easy-to-use interactive graphic interface for each function module and thereby the characteristic parameters (including PDI of hyperbranched core and linear arm unit, DP of core and arm, and DB *et al.*) can be easily defined via keyboard input. Moreover, various types of output files can be generated from HBP Builder for the direct use by DPD/CGMD and all-atom simulations, for example used by HOOMD, Tinker and Gromacs simulation packages. Due to the modular architecture of HBP Builder, it can be easier extended and reused it as a part of other programs. The design ideas and implementation process of each function module in HBP Builder will be described explicitly by building various topological structures of HBPs/HBMCs molecules. The remainder of this article is organized as follows. Section 2 respectively presents the design ideas and application examples of HBP Builder_CG for generating coarse-grained HBPs and HBMCs. In section 3, the basic principle and application examples for converting coarse-grained HBPs/HBMCs to fully atomistic structures in HBP Builder_AA module are given. The performances of HBP Builder are given in section 4. Finally, conclusion and outlook for HBP Builder are stated in section 5.

The HBP Builder_CG function module

The HBP Builder_CG module mainly realized two functions: one is to generate coarse-grained HBPs according to user required PDI, DP and DB; the other is to build HBMCs based on the generated HBPs. The generation process of HBPs and HBMCs are respectively described in more detail below.

The generation of HBPs. Unlike monodisperse polymers, HBPs are a mixture of thousands of different topological polymers. In the actual process of experiments not only the molecular weight has polydispersity, but even with same DB and DP also has many possible topological structures. So investigating the properties of HBPs through monodisperse model is obviously not appropriate. Thus, constructing polydisperse HBPs to represent the real system according to the parameters measured from experiment is one of main function of HBP Builder_CG module.

In Fig. 1a, we present the flowchart to show the generation process of coarse-grained HBPs through the HBP Builder_CG module. This flowchart was made of three subprograms: the input setting of the characteristic parameters, the coarse-grained HBPs generation, and the output of HBPs structures. The detailed descriptions for each subprogram are in the following sections.

Input setting of the characteristic parameters. The interactive graphic interface of HBP Builder_CG defines the characteristic parameters of hyperbranched system, as shown in Fig. 1b. User can control over the topological structures of the output file through it. The specific parameter setting and description are as follows.

Average degree of polymerization of the HBPs (ADP) is the average DP of the required HBPs, which should be at least greater than or equal to 3.

Required number of the HBPs (RNP) is the number of HBPs needs to be generated.

Polydisperse distribution mode (PDI_mode) and the polydispersity index (PDI) of the HBPs describe the distribution mode and the distribution width of molecular weight. In order to represent the wide molecular weight distribution in cationic polymerization processes, Schulz-Zimm and Poisson distributions were utilized in *HBP Builder_CG*, respectively, to construct our polydisperse hyperbranched systems. The Schulz-Zimm distribution obeys the form of^{54–56}:

$$p(DP) = \frac{u^u \delta^{u-1} \exp(-u\delta)}{ADP\Gamma(u)} \quad (1)$$

where $\delta = DP/ADP$. The PDI for this distribution is $PDI = (u+1)/u$. The Poisson distribution is specified by⁵⁴:

$$p(DP) = \frac{\exp(1 - ADP)(ADP - 1)^{DP-1}}{(DP - 1)!} \quad (2)$$

the PDI for Poisson distribution is $PDI = 1 + (ADP - 1)/ADP^2$. As a menu setting in *HBP Builder_CG*, user can freely choose Schulz-Zimm or Poisson distributions for HBPs.

Branch number of the root node (BR) is the graft number of the root node, which should be greater than 1.

Degree of branching of the HBPs (DB) describes the statistical DB of HBPs.

Moreover, generally speaking, DP and DB are two characteristic parameters to describe a specific hyperbranched structure, which is defined as⁵⁷:

$$DB = \frac{D + T}{D + L + T} = \frac{DP - L}{DP}; \quad DP = D + L + T \quad (3)$$

In Equation (3), D, L and T are the number of dendritic unit, the number of linear unit and the number of terminal unit respectively. In fact, only DB and DP cannot fully describe the connectivity of HBPs, because it is obviously that one HBP has many connection ways for same DB and DP. Therefore, we also employed Wiener index (WI)⁵⁶ to characterize different hyperbranched structures in our program. WI is defined as the cumulative distance between all pairs of units within the molecules measured in terms of the number of intervening bonds between two units. Herein it is calculated according to the following expression^{58,59}.

$$WI = \sum_{s=1}^{Nbonds} (V_{L,s} \times V_{R,s}) \quad (4)$$

In Equation (4), $V_{L,s}$ and $V_{R,s}$ are the number of units in the left and that in the right of the center of each bond s . As reported by Karatasos *et al.*⁵⁹, for each DB and DP, low WI describes a relatively compact dendrimer-like structure and one with a relatively high WI describes a less compact, closer to a star-like shape. In other words, structures with lower WI seem to develop their branches closer to their inner core, while structures with higher WI are branched toward the periphery. Once *HBP Builder_CG* run has finished, it will list all the structures' WI in the output file, user can select the interested structures for further investigation.

The generation process of Coarse-grained HBPs and output. As shown in Fig. 1a, in the practical generation process, we firstly calculated the number of HBPs needs to be generated for each DP according to the ADP, PDI and RNP. Then the subprogram of coarse-grained structure generation will be called to build HBPs for each DP. In this section, we will introduce the concrete realization of coarse-grained HBP generation.

Based upon the graft number of the root node defined in the input window, *HBP Builder_CG* initializes root unit firstly. For example, if assign 2 to the branchpoint of root node, it will generate a linear structure. Otherwise it will generate a branched structure. As shown in Fig. 2a, the root unit has several possible states. However, initialize which one depended on the BR value of user's input. After the initialization of root node is complete, we built HBPs with desired DP from the initial unit. The added process of the next DP-1-BR units will accord to the specific random judgment condition. The random number will be generated through the random event in the computer. The addition process of new units is from inner to the exterior and the new units are added one after another until the required DP is reached. In order to clearly explain the added rule utilized in *HBP Builder_CG* module, we take Fig. 2b as an example to illustrate. Firstly, two chances to generate and judge random number were given to each to be connected unit. Then there are three possible cases appear after that: (a) successfully added two new units; (b) successfully added one new unit; (c) no new unit added. For the first two cases, *HBP Builder_CG* will go to the next unit and repeat this addition process. But for the last case, *HBP Builder_CG* will give this unit another two added chances until at least one new unit successfully added in this position. It should be noted that we have done a good deal of work about the method to judge new unit addition. After a lot of attempts, we classify the judgment conditions into two cases: DB greater than 0.5 and DB less than or equal to 0.5. And different approaches are used in different cases. The concrete realization in the program is as follows:

```
if (db <= 0.5 && db >= 0) randb = (int)/(1.0/db);
else if (db > 0.5 && db <= 1.0) randb = (int)/(1.0/(1.0-db));
if ((db > 0.5 && rand()%randb != 0) || (db <= 0.5 && rand()%randb = 0) added a new node...
```

A large number of trails have shown that the above judgment method for generating required structure is effective and time-saving. Next, we compute DB of the generated structure when DP reaches the set value. If the deviation

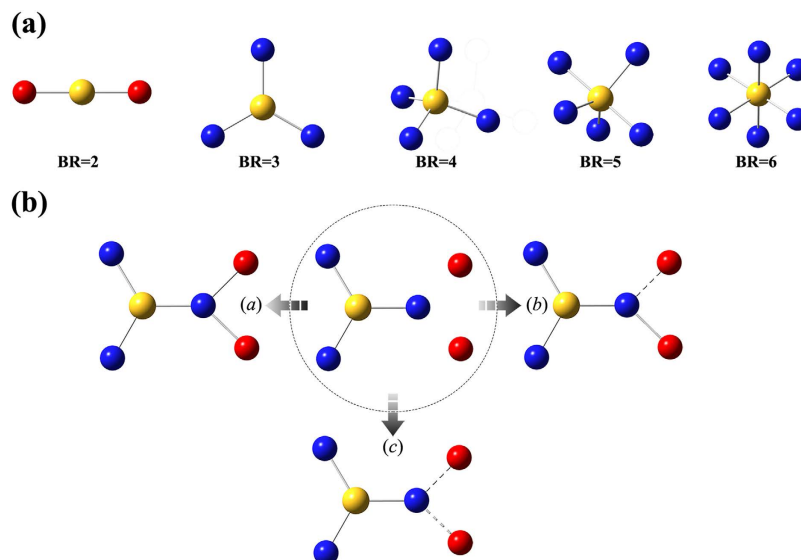


Figure 2. The initialization of the root node for linear and branched structures. (a) The diagram of the addition rules (b). The full line represents successfully added; the dotted line represents unsuccessfully added. Yellow bead represents root unit; blue bead represents the initialized units; red bead represents the new added units.

between the real DB and that of the set value is in an acceptable range (user can set the tolerance value in the program), *HBP Builder_CG* will save the coarse-grained structure. Otherwise, if the deviation exceeds the range of accuracy or the number of required HBPs is not reached, *HBP Builder_CG* will repeat the above generation process.

It is important to note that a method to judge repetitive structure is required when the number of HBPs is greater than 1. Herein, an unordered tree-inclusion subtree-matching model was employed in *HBP Builder_CG* program to remove the repetitive structure. In the matching model, each HBP could be seen as an unordered tree grows from the root node. And each unit of HBPs can be viewed as a node of the tree. Then each subtree of two unordered tree will be compared by using a recursive method. If there is a one-to-one correspondence between each subtree, it means that these two structures are equivalent. Then the information of the repetitive structure will be discarded and the above generation process will be executed repeatedly. Through the repeated verification, the unordered tree-inclusion matching model has proved to be a precise and efficient way for discarding the repetitive structures.

After all the required HBPs has been successfully generated, *HBP Builder_CG* will save molecular structure information for each HBP and packed all the molecules into one simulation box. Furthermore, the solvent molecules can be added by calling *PACKMOL* software⁶⁰. Finally, we respectively saved the output file to xml type for DPD simulation (employed by *HOOMD* program package) and gro (and itp) type for coarse-grained simulation (employed by *Gromacs* program package).

The application examples for Coarse-grained HBPs. To illustrate the practicality of *HBP Builder* for generating HBPs, a series of examples were presented to illustrate the main usage and characteristics of *HBP Builder* package. In fact, it is very simple to get the *HBP Builder* running with the command line.

```
python hbpbuilder-cg.py
```

Then it will pop up a dialogue as shown in Fig. 1b and all the parameters can be set through it. After all parameters setting was finished, user should press save button to save these input parameters and then press run button to run the program. Figure 3 present several examples generated by *HBP Builder_CG*, from which we can find *HBP Builder_CG* has several features:

- (1) HBPs is a mixture of many HBP molecules with a distribution of topological structure. As shown in Fig. 3a, *HBP Builder_CG* can quickly generate many HBPs with variable topological structures but with the same DP, DB, and BR. Based on this, we can investigate the effect of the topological structure on the property of individual HBP or the mixture of them.
- (2) In experiment, chemists can synthesize HBPs from the core molecules with multi functionalities, and the functionalities of the core molecules was defined as BR in our *HBP Builder*. By using *HBP Builder_CG*, we can generate HBPs with the same DP and DB, but with different BR (Fig. 3b). Based on this, we can investigate the effect of the core molecule structure on the property of individual HBP or the mixture of them.
- (3) It is well-known that the physicochemical properties of HBPs, including the rheological properties, crystallization and melting behaviors, glass transition, thermal and hydrolytic degradation, optoelectronic properties, and so on, are highly dependent on DB. *HBP Builder_CG* can generate various structures with the same DP and BR, but with different DB (Fig. 3c). Based on these structures, we can furtherly disclose the relationship between DB and the properties of HBPs from a theoretical point of view.

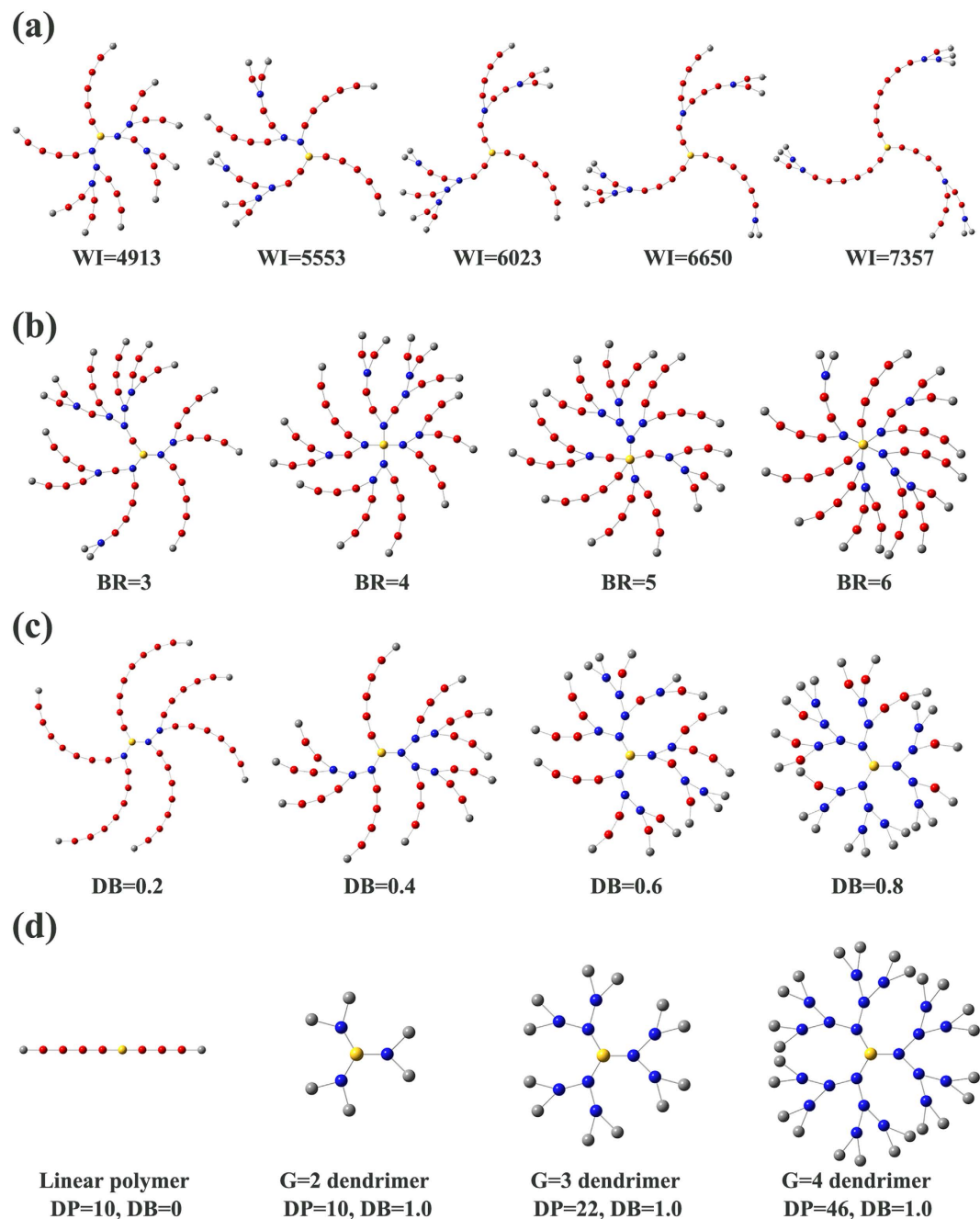


Figure 3. Different topological structures of HBPs generated by *HBP Builder_CG* module. Five topological structures of HBP with same DP ($DP=40$), DB ($DB=0.4$), BR ($BR=3$) and different WI (a); Four topological structures with same DP ($DP=60$), DB ($DB=0.4$), and different BR (b); Four topological structures with same DP ($DP=50$) and different DB (c); Linear polymer and dendrimers (d). Yellow bead represents root unit; blue bead represents dendritic unit; red bead represents linear unit; gray bead represents terminal unit.

(4) In addition to the irregular HBPs, *HBP Builder_CG* also can quickly build well-defined structures, including linear polymers ($BR=2$, $DB=0$) with different compositions and molecular weight, and dendrimers ($DB=1.0$) with different generations (Fig. 3d).

The generation of Coarse-grained HBMCs. HBMCs are composed of hyperbranched core and many linear arms, and are more complex than HBPs since the grafted linear arms also have polydispersity. Thus, the arm unit generation part was added to the *HBP Builder_CG* module to control the generation and the linking of arm units. In Fig. 4a, we present a flowchart to show the generation process of coarse-grained HBMCs through the *HBP Builder_CG* module. Likewise, an interactive graphic interface was also developed for building HBMCs (Fig. 4b). In this section, the characteristic parameters for describing linear arm units and the generation and linking process of arm units will be introduced explicitly.

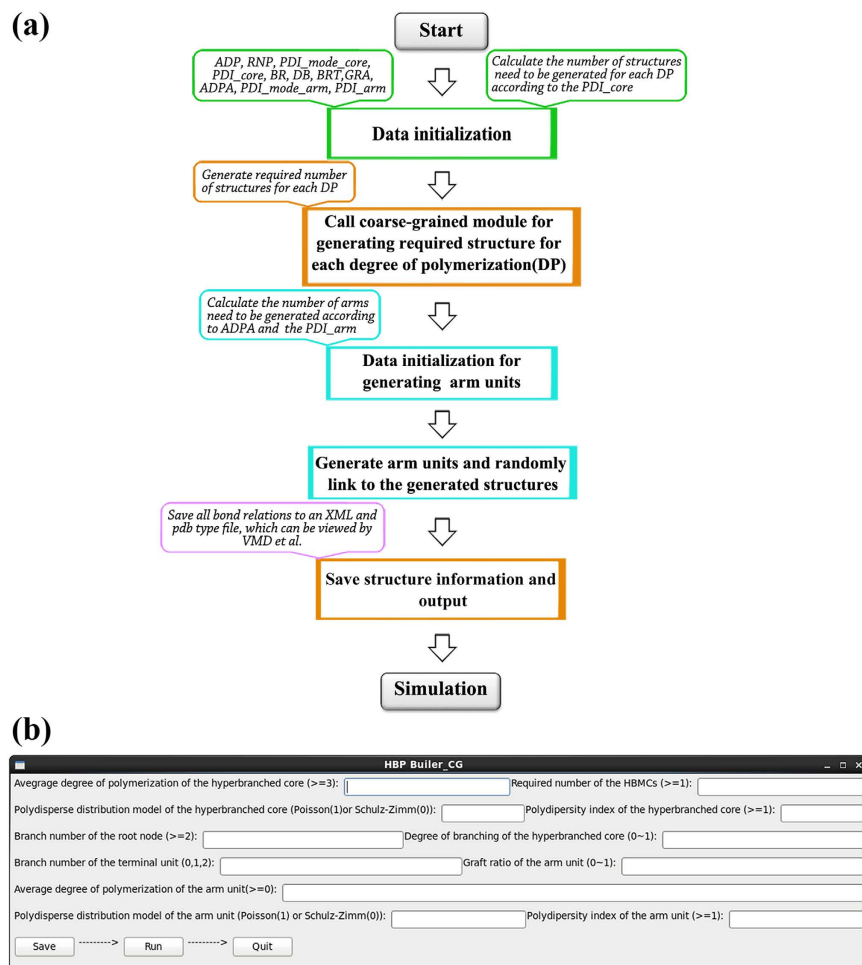


Figure 4. The flowchart of *HBP Builder_CG* for generating HBMCs. (a) The interactive input graphic interface of *HBP Builder_CG* for generating HBMCs (b).

Input setting of the characteristic parameters. As shown in Fig. 4b, there are several characteristic parameters for arm unit need to be set except the parameters for hyperbranched core. The specific parameters added for HBMCs are as follows.

Branch number of the terminal units (BRT) is the number of the allowed linking position of each terminal unit.

Graft ratio of the arm units (GRA) is the ratio of the arm units bonded to terminal and linear units. If assign zero to *GRA*, HBMCs molecules without arms will be generated, which returns back to HBPs.

Average degree of polymerization of linear arm (ADPA) is the average DP of the arm units (DP_{arm}).

Polydisperse distribution mode (PDI_mode_arm) and the polydispersity index (PDI_arm) of the linear arms are the distribution mode and the width of the molecular weight of the arm units. Schulz-Zimm and Poisson distributions were also available for arm units.

The generation and output of coarse-grained HBMCs. As shown in Fig. 4a, the generation process of coarse-grained HBMCs is based on the generation method of HBPs. That is, the generation process for hyperbranched core of HBMCs is identical to that of HBPs, which are not repeated here. And on this basis, after all required hyperbranched core generation has completed, the arm building subprogram will be called to generate the linear arm units with required number and length. Similar to the core generation, we firstly calculated the number of arms needs to be generated for each molecular weight according to the ADPA, PDI_mode_arm, PDI_arm and GRA of the arm units. Then all arm units will be generated one by one, and next randomly link it with the allowed joint units of the hyperbranched core one by one. After all arms are completely linked, we save molecular structure information for each HBMCs and packed all the molecules into one simulation box. Furthermore, the solvent molecules can be added in the same way as it used for HBPs. Finally, the xml and gro (and itp) type file were respectively outputted for DPD and coarse-grained molecular simulations (employed by Gromacs program package).

The application example of coarse-grained HBMCs. As shown in Fig. 5 and Figure S2 (supplementary information), we also present a series of examples to show the generated structures of HBMCs. It can be seen

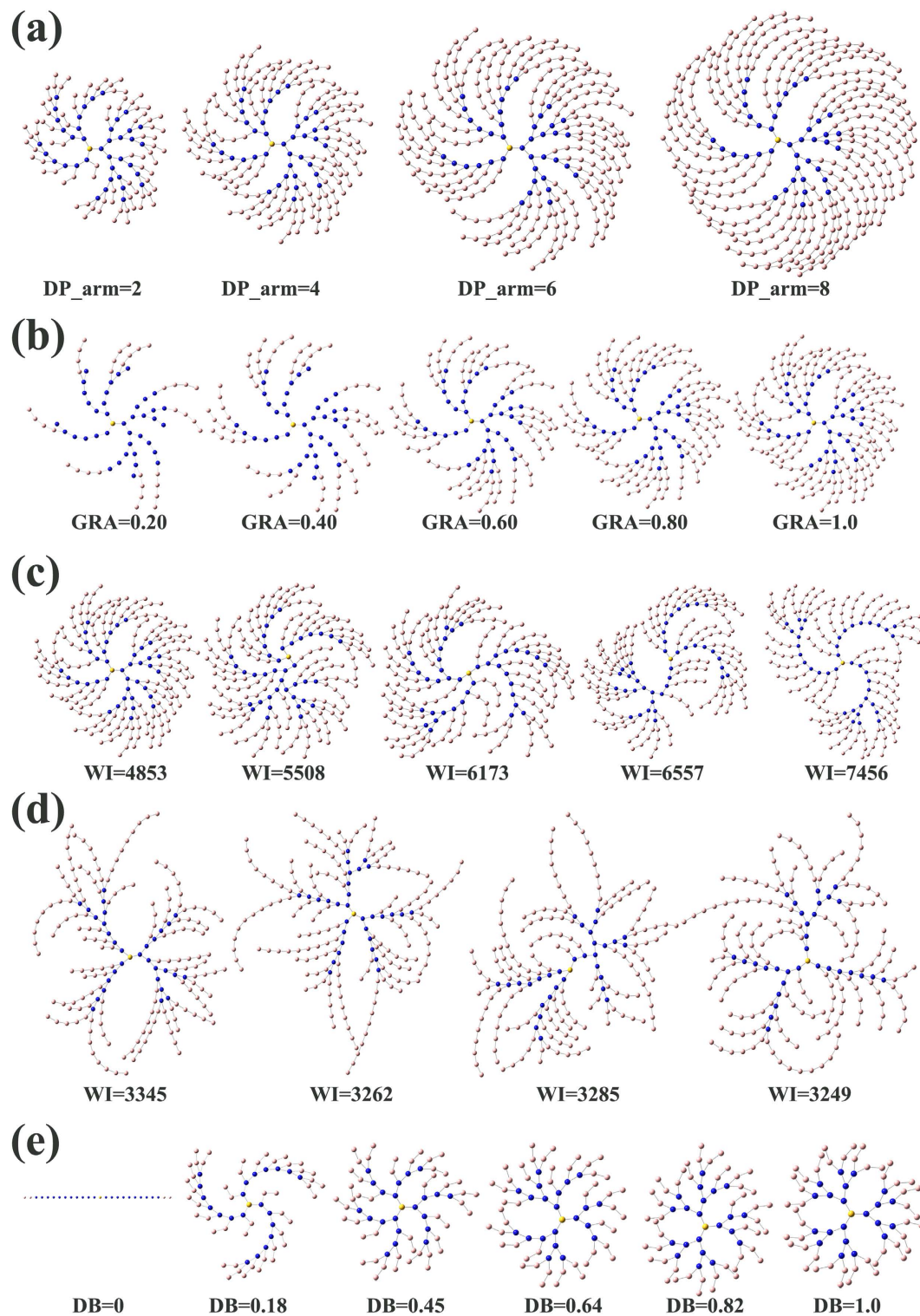


Figure 5. Different topological structures of HBMCs generated by *HBP Builder_CG* module. Four topological structures with same DP ($DP = 40$) of hyperbranched core and different arm length of linear arm (DP_{arm}) (a); Five topological structures with same hyperbranched core ($DP = 40$, $DB = 0.40$, $DP_{arm} = 4$) and different GRA (b); Five topological structures with same DP ($DP = 40$), DB ($DB = 0.40$), DP_{arm} ($DP_{arm} = 4$) and different WI of the hyperbranched core (c); Four topological structures with same DP ($DP = 32$), DB in the core ($DB = 0.38$), $ADPA$ ($ADPA = 5$) and different arm length distribution and WI in the core (d); Six topological structures with same DP ($DP = 22$), DP_{arm} ($DP_{arm} = 2$) and different DB in the core ($DB = 0$ is a ABA triblock copolymer, $DB = 1.0$ is a dendritic multi-arm copolymer with a dendrimer core) (e). Yellow bead represents root unit; blue bead represents dendritic unit and linear unit of the hyperbranched core; pink bead represents linear arm units.

that the arm length, the graft ratio and PDI can be controlled flexibly. And the polydisperse HBMCs can be built conveniently through *HBP Builder_CG* module. Its main application range summarized below:

- (1) Amphiphilic HBMCs have been proved to be excellent self-assembly precursors. In experiment, HBMC with different arm length was designed to study the influence of the hydrophilic and hydrophobic ratio on the HBMC self-assembly behavior. In *HBP Builder_CG*, user can quickly build various HBMC structures with the same hyperbranched core but different length of arms (Fig. 5a).
- (2) The GRA is an important index to describe HBMC structure, and is also very important to regulate the properties of HBMCs. As shown in Fig. 5b, user can build various HBMC structures with the same hyperbranched core and different arms with variable GRAs through *HBP Builder_CG*. These structures can be used to disclose the GRA effect on the properties of HBMCs.
- (3) In the course of the HBMC self-assembly, the topological structure of the hyperbranched core has an effect on the self-assemblies. By using *HBP Builder_CG*, user can generate various HBMC structures (Fig. 5c) with the same arms (*GRA* and *DP_arm* are constant) and different topological hyperbranched cores (BR, DB, DP are the same, but the topology (WI) is different). Furthermore, the topology effect on the self-assembled structures can be investigated.
- (4) In the actual process of experiment, the arm length of the HBMCs always have a wide range of distribution. By using *HBP Builder_CG*, user can generate various HBMC structures (Fig. 5d) with the same DP and DB of the hyperbranched cores, and the average arm length (ADPA is constant), but different WI of the hyperbranched cores (DP are the same) and arm length with random distribution.
- (5) DB is often used in regulating the self-assembled morphologies of HBMCs. By using *HBP Builder_CG*, user can generate various HBMC structures (Fig. 5e) with the same arms (*GRA* and *DP_arm* are constant) and different DB of the hyperbranched cores (DP are the same). We can see HBMC turn to ABA triblock copolymer when DB decreases to 0. In the meanwhile, HBMC turn to dendritic multi-arm copolymer with a dendrimer core when DB increases to 1.0. Furthermore, the DB effect on the self-assembled structures can be studied.
- (6) In experiments, in solution or bulk state, it is a mixture of many HBMC molecules with a polydispersity of molecular weight in both hyperbranched core and arms, and a polydispersity of DB and a distribution of topological structure in the hyperbranched core. It is very complex. However, by using our *HBP Builder_CG*, the user can quickly generate a lot of polydisperse HBMCs just by inputting the average parameters of PDI (for the cores and arms) and DB (for the cores) through the keyboard (Figure S2). Thus, these simulation boxes can be used to disclose the PDI effect on the self-assembly behaviors of HBMCs. The molecular weight distributions of the generated HBPs/HBMCs by *HBP Builder_CG* agree well with the theoretical predications from Schulz-Zimm or Poisson distributions (Figure S3).

The *HBP Builder_AA* function module

The *HBP Builder_AA* is comprised of three subprograms: the input of the characteristic parameters, the coarse-grained HBPs/HBMCs reading or building, and the fully atomistic HBPs/HBMCs generation and output. We present the flowchart of the *HBP Builder_AA* module in Fig. 6a, which is explained in detail in this section.

Input setting of the characteristic parameters. As shown in Fig. 6b, an interactive graphic interface was developed for *HBP Builder_AA* module. Thus all the related parameters can be set through it. The detailed notes on these parameters are explained as follows:

Except several parameters have been mentioned in *HBP Builder_CG* module, there are several characteristic parameters for describing the atomistic structure information. First of all, user should provide the name and types of all-atom repeat units. It worth noted that in some cases, several reactive place of the dendritic or the linear or the terminal units is not equal, which caused one unit has several possible connection ways, for instance, the two hydroxyl groups in the repeating unit of hyperbranched polyglycerol (HPG) are not equal, and there are two possible connection way. Thus, the possible connection ways of the repeating units with unequal reactivity should be defined in the dialogue box.

- A3_0*: the central core unit.
- A3_1*: the dendritic unit.
- A2_1*: the linear unit 1.
- A2_2*: the linear unit 2.
- A1*: the terminal unit.
- A11_1*: first type of the terminal unit with one arm unit.
- A11_2*: second type of the terminal unit with one arm unit.
- A12 type unit*: the terminal unit with two arms.
- L1 type unit*: the arm unit.

Furthermore, the file name, the file location, and the index of the atom at joint points should be specified in a map file. Then, these possible connection ways will randomly appear in the generated polymers. The format of the map file is shown in Table S1 (supplementary information) and the corresponding structure segments can be seen in Fig. 7f.

The serial number of the required all-atom structure should be greater than zero and less than or equal to the required number of HBPs/HBMCs. Here a multi-line text input was created and the format is 'N1 N2'. All

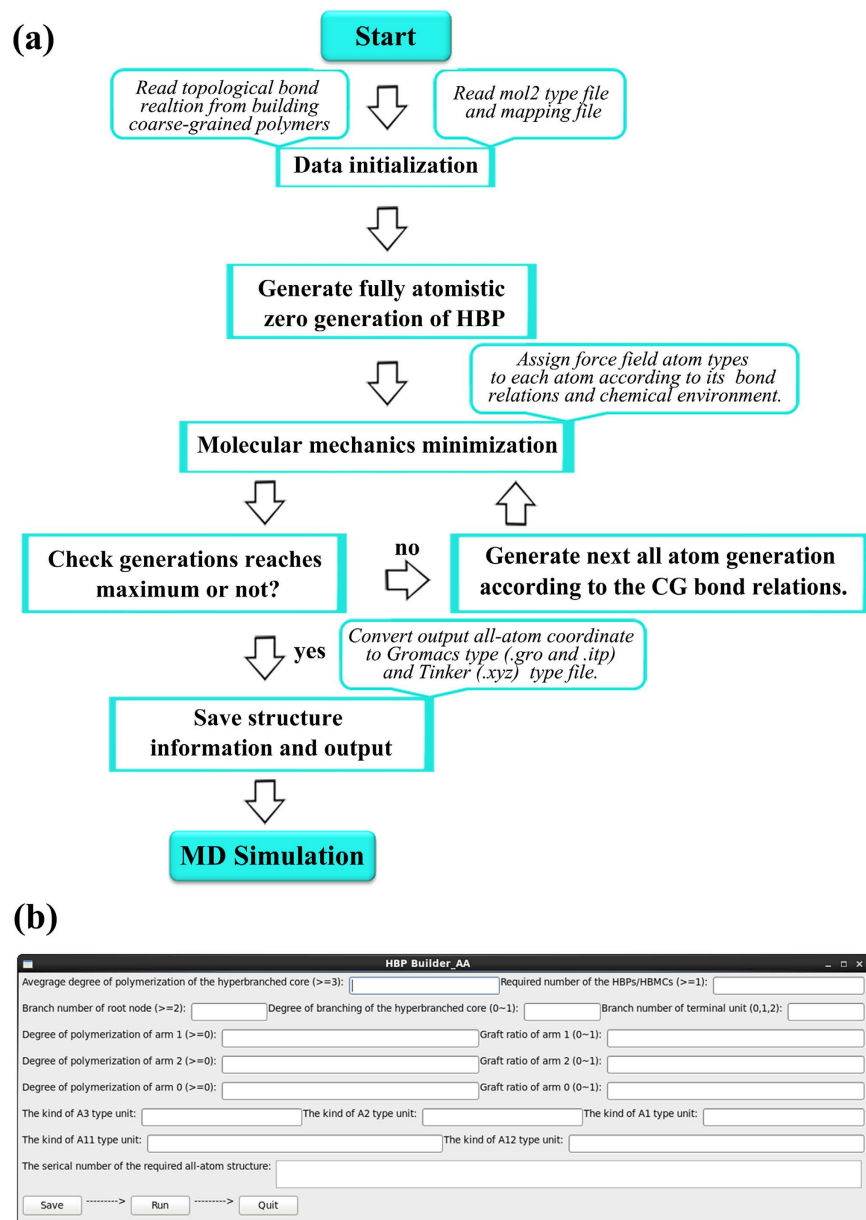


Figure 6. The flow chart of *HBP Builder_AA* module. (a) The interactive input graphic interface of *HBP Builder_AA* module (b).

structures for which index greater than or equal to N_1 and less than or equal to N_2 will be converted. If N_1 equals N_2 means only one structure required to be converted. User can add the serial number of other one or more structures in the next line.

Moreover, if HBMC with special topological structure is required, for example, considering the flexibility of the program, user can set the branch number of the terminal unit, and two arms with the same branch point can have their own component unit and graft ratio. The next several parameters need to be set:

Degree of polymerization of arm 0: the DP of the arm unit connected to linear units.

Graft ratio of arm 0: the graft ratio of the arm unit connected to linear units.

Degree of polymerization of arm 1: the DP of the arm unit connected to the first position of terminal unit.

Graft ratio of arm 1: the graft ratio of the arm unit connected to the first position of the terminal unit.

Degree of polymerization of arm 2: the DP of the arm unit connected to the second position of the terminal unit.

Graft ratio of arm 2: the graft ratio of the arm unit connected to the second position of the terminal unit.

The back mapping process of *HBP Builder_AA*. In fact, there are two sources to obtain the topological bond relations in *HBP Builder_AA* module. One of them is read it from the generated coarse-grained HBP molecules. The other is build structures by their own special parameters as shown above. The coarse-grained HBPs/

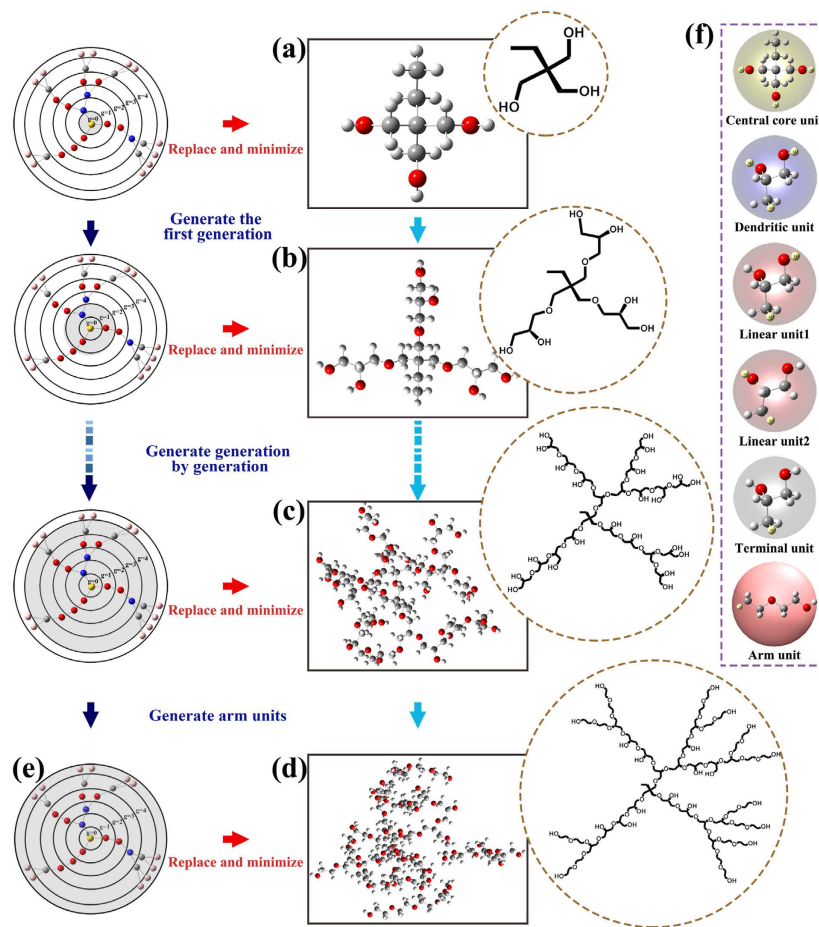


Figure 7. Scheme of the generation process of fully atomistic structure. First step for generating zero generation (a); Second step for generating first generation (b); The structure of HPG core (c); The structure of HPG-star-PEG (d) and its corresponding CG topology (e); The building units of HPG-star-PEG molecule (f).

HBMCs generation process of this module is identical to the coarse-grained generation subprogram used above. When coarse-grained topological bond relations were obtained, the fine-graining subprogram will be called to convert them to fully atomistic structure. Here a back-mapping strategy was utilized to generate fully atomistic HBPs/HBMCs molecules according to the coarse-grained bond relations and the information of all-atom repeat units. Next, in order to obtain the reasonable molecular structure, the repeated replacement and energy minimization generation by generation process were realized in *HBP Builder_AA* module. Here, it should be noted that the main idea of this module is similar to that used in DBT⁴⁰.

We take HPG-star-PEG as an example to illustrate the mapping process. The explicit process is as follows:

- (1) A coarse-grained HPG-star-PEG was generated and each CG unit was assigned to its corresponding generation number, as shown in Fig. 7e. Next the zero-generation of the HBMC molecules was replaced with the preset all-atom units (Fig. 7f). In *HBP Builder*, the preset all-atom units were written in mol2 type. Meanwhile, the hydrogen atoms which to be deleted in the linker positions were listed in the map file (the yellow atom in Fig. 7f and Table S1).
- (2) Assigning atom type of force field to each atom of the generated molecule according to the bond relations and chemical environment of each atom. Currently only a part of atom types of OPLS_AA force field has been incorporated in *HBP Builder_AA* module. Nonetheless, it does not matter, because other atom types can be easily added to *HBP Builder_AA* module. Then energy minimization of molecular mechanics was employed to the generated molecule until satisfy the convergence criterion. Next, we saved structure information of the optimized molecule for the next generation stage.
- (3) We added first fully atomistic generation to the above optimized zero-generation structure according to the coarse-grained bond relations (Fig. 7a). Meanwhile, we linked discrete all-atom units and deleted the redundant hydrogen atoms. After that, succeeding steps will be performed in order, store the atom coordinates and bond relations, assign atom types, optimize the generated structure and save the structure information (Fig. 7b). As mentioned earlier, the branchpoint of dendritic unit, linear unit and terminal unit are not equivalent in some cases, user should input all the possibility and provide the label of the hydrogen atom of the connection points. As shown in Fig. 7f, the HPG core has two possible linear units. So the coarse-grained linear units will be substituted for these two all-atom units randomly.

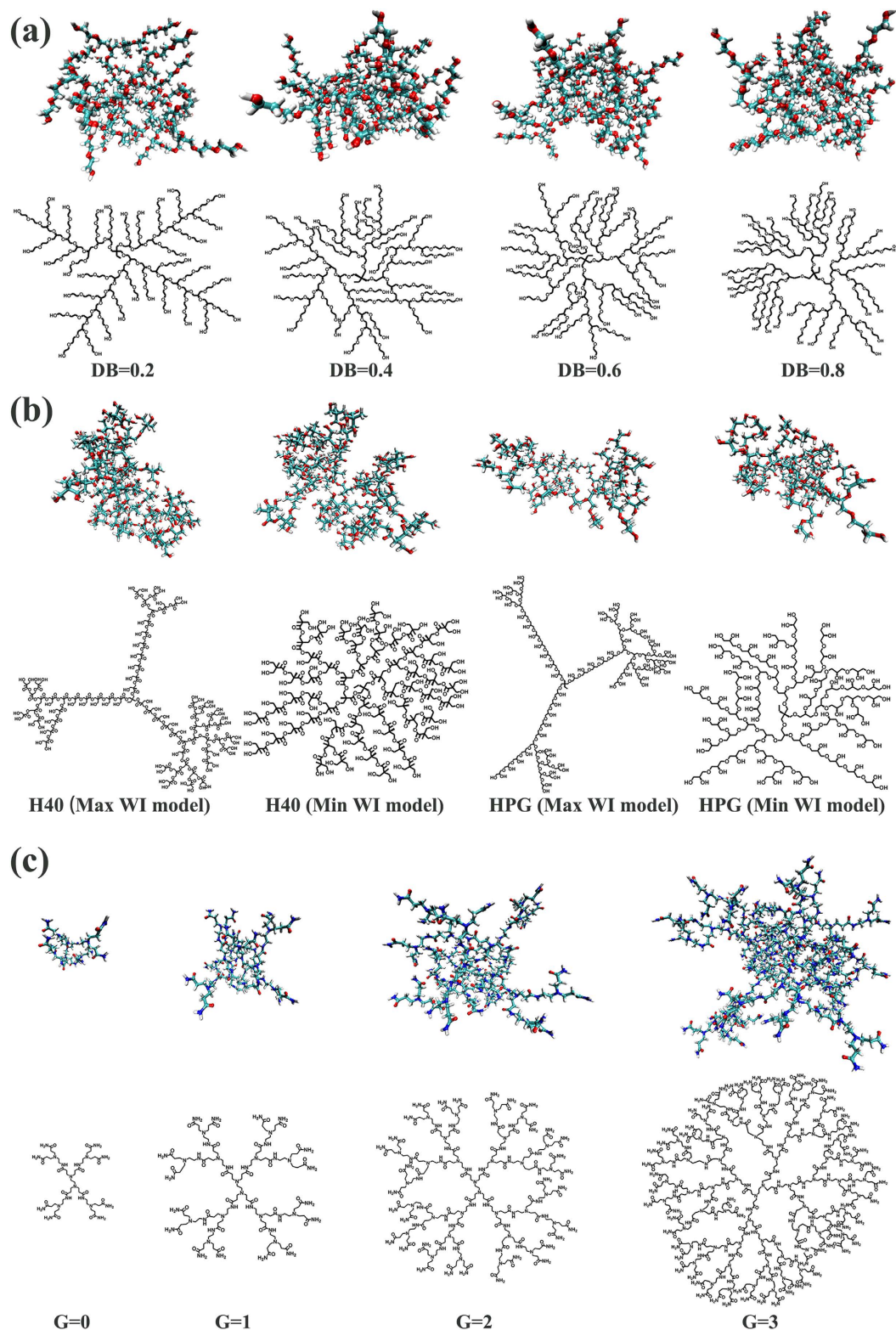


Figure 8. Different fully atomistic structures of HBPs generated by *HBP Builder_AA* module. HPG-star-PEG with different DB (a); Max WI and Min WI model molecules for H40 and HPG (b); PAMAM dendrimers with different generation (c). In each figure, the upper one is the CPK model, and the lower one is the corresponding structural formula.

- (4) Repeat the third step until the entire HPG core was completely replaced and optimized (Fig. 7c).
- (5) Then, the coarse-grained arm unit will be replaced with the PEG unit. The process is similar to that generation process of HPG core, we link discrete all-atom units and delete the redundant hydrogen atoms, assign

- atom types, optimize the generated structure and save the structure information (Fig. 7d).
- (6) Finally, two types of files were outputted for MD simulations after the structure generation. They are the xyz type file for Tinker program and gro (and itp) type file for Gromacs package. User can add additional output file types to meet their own needs easily by minor revisions of the program. In order to make the output more intuitively, the graphic library OpenGL was linked in *HBP Builder* program. Thus the final structure can be directly observed through the display window. Of course, user can also view it through other visualization software such as VMD etc.³⁸.

The application example of *HBP Builder_AA*. The atomistic insight is very important when special interactions need to be considered, for example, the noncovalent interactions like hydrogen bonds or specific chemical interactions between polymers, drugs, DNAs or proteins. Here, to show the fully atomistic structures generated through *HBP Builder_AA* module, a series of examples were presented.

- (1) User can build a series of fully atomistic HBMC structures with different *DB* values in the hyperbranched core. As shown in Fig. 8a, several HPG-star-PEGs with different *DB* values were generated through *HBP Builder_AA*. Furthermore, these model molecules can be used to study the dependency of unimolecular properties on the *DB*s.
- (2) User can build different fully atomistic HBPs based the same coarse-grained structures. As shown in Fig. 8b, two fully atomistic structures for H40 and HPG molecules were generated through *HBP Builder_AA*. Among them, Max WI models of H40 and HPG (or Min WI model of H40 and HPG) have the same coarse-grained structure, however, the fully atomistic structures are totally different. One is the hyperbranched polyester (H40), and the other is hyperbranched polyether (HPG). On the basis of these structures, user can investigate the properties of different type of all-atom HBP molecules with same coarse-grained topology.
- (3) User also can build fully atomistic structures for regular dendrimers. The *HBP Builder_CG* can generate coarse-grained dendrimers. Then after inputting the repeat unit information, as shown in Fig. 8c, the fully atomistic structures of PAMAM dendrimers from zeroth to third generation were built through *HBP Builder_AA*. Thus, the properties of these dendrimers can be studied through molecular simulation.

The performance of *HBP Builder*

To illustrate the high efficiency of *HBP Builder*, Table S2 (supplementary information) summarizes several systems and the corresponding computational time required to generate them. All these examples were run on an Intel(R) Core(TM) Quad i7-3610QM CPU with 2.30 GHz clock rate and 8 Gb memory, and implemented under the openSUSE 13.2 embedded operating system. Table S2 shows that with the increment of *DP* and *RNP*, the computational time is increased gradually. Overall, however, *HBP Builder* has a high efficiency to generate the user required structures.

Conclusions and Outlook

In this work, *HBP Builder* program for building HBPs/HBMCs with different degree of polymerization, degree of branching and polydispersity index is presented. Several parameters for characterizing the features of hyperbranched molecules are introduced to control over the final topological structures. An unordered tree-inclusion matching model is used in the program to eliminate the repetitive structures. Furthermore, fully atomistic structures are obtained through a back-mapping scheme, which generated and optimized the HBPs/HBMCs structures generation by generation. We have generated a series of coarse-grained and fully atomistic structures to illustrate the application and validation of *HBP Builder*. These results have been shown that our tool is very useful and can be further extended to build HBPs/HBMCs of any desired degree of polymerization, degree of branching and polydispersity index by providing structure parameters and user defined building units. In the meanwhile, *HBP Builder* can also be used to build linear block copolymers, dendrimers or dendritic multi-arm copolymers when set degree of branching in the cores to 0 or 1.0. The modular architecture making *HBP Builder* easier to add new molecule types and force fields and also reuse it as a part of other programs. We have reasons to believe *HBP Builder* will be a promising tool to construct model molecules for studying of HBPs/HBMCs interacts with the polymers, drug molecules, protein and nucleic acid, and HBMCs self-assembly in solution.

References

1. Whitesides, G. M. & Grzybowski, B. Self-Assembly at All Scales. *Science* **295**, 2418–2421 (2002).
2. Wang, F., Wilson, M. S., Rauh, R. D., Schotland, P. & Reynolds, J. R. Electroactive and Conducting Star-Branched Poly(3-hexylthiophene)s with a Conjugated Core. *Macromolecules* **32**, 4272–4278 (1999).
3. Wang, F., Rauh, R. D. & Rose, T. L. An Electrically Conducting Star Polymer. *J. Am. Chem. Soc.* **119**, 11106–11107 (1997).
4. Gao, C. & Yan, D. Hyperbranched polymers: from synthesis to applications. *Prog. Polym. Sci.* **29**, 183–275 (2004).
5. Voit, B. I. & Lederer A. Hyperbranched and Highly Branched Polymer Architectures—Synthetic Strategies and Major Characterization Aspects. *Chem. Rev.* **109**, 5924–5973 (2009).
6. Hawker, C. J., Lee, R. & Frechet, J. M. J. One-Step Synthesis of Hyperbranched Dendritic Polyesters. *J. Am. Chem. Soc.* **113**, 4583–4588 (1991).
7. Kim, Y. H. & Webster, O. W. Water-Soluble Hyperbranched Polyphenylene: “A Unimolecular Micelle”? *J. Am. Chem. Soc.* **112**, 4592–4593 (1990).
8. Uhrich, K. E., Hawker, C. J., Frechet, J. M. J. & Turner, S. R. One-Pot Synthesis of Hyperbranched Polyethers. *Macromolecules* **25**, 4583–4587 (1992).
9. Bolton, D. H. & Wooley, K. L. Synthesis and Characterization of Hyperbranched Polycarbonates. *Macromolecules* **30**, 1890–1896 (1997).

10. Hawker, C. J. & Chu, F. Hyperbranched Poly(ether ketones): Manipulation of Structure and Physical Properties. *Macromolecules* **29**, 4370–4380 (1996).
11. Ishida, Y., Sun, A. C. F., Jikei, M. & Kakimoto, M.-A. Synthesis of Hyperbranched Aromatic Polyamides Starting from Dendrons as AB_x Monomers: Effect of Monomer Multiplicity on the Degree of Branching. *Macromolecules* **33**, 2832–2838 (2000).
12. Claesson, H. *et al.* Rheological behaviour during UV-curing of a star-branched polyester. *Prog. Org. Coat.* **44**, 63–67 (2002).
13. Staring, E., Dias, A. A. & Benthem, R. A. T. M. V. New challenges for R&D in coating resins. *Prog. Org. Coat.* **45**, 101–117 (2002).
14. Schmaljohann, D. *et al.* Blends of Amphiphilic, Hyperbranched Polyesters and Different Polyolefins. *Macromolecules* **32**, 6333–6339 (1999).
15. Jang, J., Oh, J. H. & Moon, S. I. Crystallization Behavior of Poly(ethylene terephthalate) Blended with Hyperbranched Polymers: The Effect of Terminal Groups and Composition of Hyperbranched Polymers. *Macromolecules* **33**, 1864–1870 (2000).
16. Zhou, Y., Huang, W., Liu, J., Zhu, X. & Yan, D. Self-Assembly of Hyperbranched Polymers and Its Biomedical Applications. *Adv. Mater.* **22**, 4567–4590 (2010).
17. Liu, Y. *et al.* A Supramolecular Janus Hyperbranched Polymer and Its Photoresponsive Self-Assembly of Vesicles with Narrow Size Distribution. *J. Am. Chem. Soc.* **135**, 4765–4770 (2013).
18. Tao, W. *et al.* A Linear-Hyperbranched Supramolecular Amphiphile and Its Self-Assembly into Vesicles with Great Ductility. *J. Am. Chem. Soc.* **134**, 762–764 (2012).
19. Zhou, Y. & Yan, D. Real-Time Membrane Fusion of Giant Polymer Vesicles. *J. Am. Chem. Soc.* **127**, 10468–10469 (2005).
20. Zhou, Y. & Yan, D. Supramolecular Self-Assembly of Giant Polymer Vesicles with Controlled Sizes. *Angew. Chem. Int. Ed.* **43**, 4896–4899 (2004).
21. Jiang, W., Zhou, Y. & Yan, D. Hyperbranched polymer vesicles: from self-assembly, characterization, mechanisms, and properties to applications. *Chem. Soc. Rev.* **44**, 3874–3889 (2015).
22. Jin, H., Huang, W., Zhu, X., Zhou, Y. & Yan, D. Biocompatible or biodegradable hyperbranched polymers: from self-assembly to cytomimetic applications. *Chem. Soc. Rev.* **41**, 5986–97 (2012).
23. Bankura, A., Karmakar, A., Carnevale, V., Chandra, A. & Klein, M. L. Structure, Dynamics, and Spectral Diffusion of Water from First-Principles Molecular Dynamics. *J. Phys. Chem. C* **118**, 29401–29411 (2014).
24. Jonchiere, R., Seitsonen, A. P., Ferlat, G., Saitta, A. M. & Vuilleumier, R. Van der Waals effects in ab initio water at ambient and supercritical conditions. *J. Chem. Phys.* **135**, 154503 (2011).
25. Chen, W., Shi, C., MacKerell, A. D. & Shen, J. Conformational Dynamics of Two Natively Unfolded Fragment Peptides: Comparison of the AMBER and CHARMM Force Fields. *J. Phys. Chem. B* **119**, 7902–7910 (2015).
26. Murzyn, K., Bratek, M. & Pasenkiewicz-Gierula, M. Refined OPLS All-Atom Force Field Parameters for n-Pentadecane, Methyl Acetate, and Dimethyl Phosphate. *J. Phys. Chem. B* **117**, 16388–16396 (2013).
27. Siu, S. W. L., Pluhackova, K. & Böckmann, R. A. Optimization of the OPLS-AA Force Field for Long Hydrocarbons. *J. Chem. Theory Comput.* **8**, 1459–1470 (2012).
28. Pol-Fachin, L., Rusu, V. H., Verli, H. & Lins, R. D. GROMOS 53A6GLYC, an Improved GROMOS Force Field for Hexopyranose-Based Carbohydrates. *J. Chem. Theory Comput.* **8**, 4681–4690 (2012).
29. Jorgensen, W. L. & Schyman, P. Treatment of Halogen Bonding in the OPLS-AA Force Field: Application to Potent Anti-HIV Agents. *J. Chem. Theory Comput.* **8**, 3895–3901 (2012).
30. Tapia, O. & Velázquez, I. Molecular Dynamics Simulations of DNA with Protein's Consistent GROMOS Force Field and the Role of Counterions' Symmetry. *J. Am. Chem. Soc.* **119**, 5934–5938 (1997).
31. Uusitalo, J. J., Ingólfsson, H. I., Akhshi, P., Tieleman, D. P. & Marrink, S. J. Martini Coarse-Grained Force Field: Extension to DNA. *J. Chem. Theory Comput.* **11**, 3932–3945 (2015).
32. López, C. A., Sovova, Z., van Eerden, F. J., de Vries, A. H. & Marrink, S. J. Martini Force Field Parameters for Glycolipids. *J. Chem. Theory Comput.* **9**, 1694–1708 (2013).
33. Levine, B. G. *et al.* Micellization Studied by GPU-Accelerated Coarse-Grained Molecular Dynamics. *J. Chem. Theory Comput.* **7**, 4135–4145 (2011).
34. Shinoda, W., Devane, R. & Klein, M. L. Multi-property fitting and parameterization of a coarse grained model for aqueous surfactants. *Mol. Simul.* **33**, 27–36 (2007).
35. Orsi, M., Ding, W. & Palaiokostas, M. Direct Mixing of Atomistic Solutes and Coarse-Grained Water. *J. Chem. Theory Comput.* **10**, 4684–4693 (2014).
36. Wassenaar, T. A., Ingólfsson, H. I., Prieß, M., Marrink, S. J. & Schäfer, L. V. Mixing MARTINI: Electrostatic Coupling in Hybrid Atomistic–Coarse-Grained Biomolecular Simulations. *J. Phys. Chem. B* **117**, 3516–3530 (2013).
37. Fritsch, S., Junghans, C. & Kremer, K. Structure Formation of Toluene around C60: Implementation of the Adaptive Resolution Scheme (AdResS) into GROMACS. *J. Chem. Theory Comput.* **8**, 398–403 (2012).
38. Humphrey, W. A., Dalke, A. & Schulten, K. VMD - Visual Molecular Dynamics. *J. Molec. Graphics* **14**, 33–38 (1996).
39. Yuan, F. & Larson, R. G. Multiscale Molecular Dynamics Simulations of Model Hydrophobically Modified Ethylene Oxide Urethane Micelles. *J. Phys. Chem. B* **119**, 12540–12551 (2015).
40. Maingi, V., Jain, V., Bharatam, P. V. & Maiti, P. K. Dendrimer Building Toolkit: Model Building and Characterization of Various Dendrimer Architectures. *J. Comput. Chem.* **33**, 1997–2011 (2012).
41. Wang, Y. *et al.* Dissipative Particle Dynamics Simulation Study on Vesicles Self-Assembled from Amphiphilic Hyperbranched Multiarm Copolymers. *Chem. Asian J.* **9**, 2281–2288 (2014).
42. Wang, Y., Li, B., Zhou, Y., Lu, Z. & Yan, D. Dissipative particle dynamics simulation study on the mechanisms of self-assembly of large multimolecular micelles from amphiphilic dendritic multiarm copolymers. *Soft Matter* **9**, 3293–3304 (2013).
43. Widmann, A. H. & Davies, G. R. Simulation of the intrinsic viscosity of hyperbranched polymers with varying topology. I. Dendritic polymers built by sequential addition. *Comput. Theor. Polym. Sci.* **8**, 191–199 (1998).
44. Lyulin, A. V., Adolf, D. B. & Davies, G. R. Computer Simulations of Hyperbranched Polymers in Shear Flows. *Macromolecules* **34**, 3783–3789 (2001).
45. Dalakoglou, G. K., Karatasos, K., Lyulin, S. V. & Lyulin, A. V. Shear-induced effects in hyperbranched-linear polyelectrolyte complexes. *J. Chem. Phys.* **129**, 034901 (2008).
46. Neelov, I. M. & Adolf, D. B. Brownian Dynamics Simulation of Hyperbranched Polymers under Elongational Flow. *J. Phys. Chem. B* **108**, 7627–7636 (2004).
47. Mulder, T., Lyulin, A. V., Schoot, P. v. d. & Michels, M. A. J. Architecture and Conformation of Uncharged and Charged Hyperbranched Polymers: Computer Simulation and Mean-Field Theory. *Macromolecules* **38**, 996–1006 (2005).
48. Wiener, H. Structural Determination of Paraffin Boiling Points. *J. Am. Chem. Soc.* **69**, 17–20 (1947).
49. Sheridan, P. F., Adolf, D. B., Lyulin, A. V., Neelov, I. & Davies, G. R. Computer simulations of hyperbranched polymers: The influence of the Wiener index on the intrinsic viscosity and radius of gyration. *J. Chem. Phys.* **117**, 7802–7812 (2002).
50. Schmidt, R. R., Cifre J. G. H. & Torre J. G. d. l. Multi-Scale Simulation of Hyperbranched Polymers. *Polymers* **7**, 610–628 (2015).
51. Echenique, R. D. R., Schmidt R. R., Freire J. J., Cifre J. G. H. & Torre J. G. d. l. A Multiscale Scheme for the Simulation of Conformational and Solution Properties of Different Dendrimer Molecules. *J. Am. Chem. Soc.* **131**, 8548–8556 (2009).
52. Shi, Z., Zhou, Y. & Yan, D. Facile Fabrication of pH-Responsive and Size-Controllable Polymer Vesicles From a Commercially Available Hyperbranched Polyester. *Macromol. Rapid Commun.* **29**, 412–418 (2008).
53. Zhou, Y. & Yan, D. Real-Time Membrane Fission of Giant Polymer Vesicles. *Angew. Chem. Int. Ed.* **44**, 3223–3226 (2005).

54. Li, Y., Qian, H.-J., Lu, Z.-Y. & Shi, A.-C. Enhancing composition window of bicontinuous structures by designed polydispersity distribution of ABA triblock copolymers. *Polymer*. **54**, 6253–6260 (2013).
55. Lynd, N. A., Hillmyer, M. A. & Matsen, M. W. Theory of Polydisperse Block Copolymer Melts: Beyond the Schulz-Zimm Distribution. *Macromolecules* **41**, 4531–4533 (2008).
56. Zimm, B. H. Apparatus and Methods for Measurement and Interpretation of the Angular Variation of Light Scattering: Preliminary Results on Polystyrene Solutions. *J. Chem. Phys.* **16**, 1099–1116 (1948).
57. Hawker, C. J., Lee, R. & Fréchet, J. M. J. One-Step Synthesis of Hyperbranched Dendritic Polyesters. *J. Am. Chem. Soc.* **113**, 4583–4588 (1991).
58. Widmann, A. H. & Davies, G. R. Simulation of the intrinsic viscosity of hyperbranched polymers with varying topology. I. Dendritic polymers built by sequential addition. *Theor. Polym. Sci.* **8**, 191–199 (1998).
59. Dalakoglou, G. K., Karatasos, K., Lyulin, S. V. & Lyulin, A. V. Effects of topology and size on statics and dynamics of complexes of hyperbranched polymers with linear polyelectrolytes. *J. Chem. Phys.* **127**, 214903 (2007).
60. Martínez, L., Andrade, R., Birgin, E. G. & Martínez, J. M. Packmol: A Package for Building Initial Configurations for Molecular Dynamics Simulations. *J. Comput. Chem.* **30**, 2157–2164 (2009).

Acknowledgements

We thank the National Basic Research Program (2013CB834506), China National Funds for Distinguished Young Scholar (21225420), the National Natural Science Foundation of China (21404070, 21474062, 91127047), and the “Shu Guang” project supported by Shanghai Municipal Education Commission and Shanghai Education Development Foundation (13SG14) for financial support. Thanks for the computing service provided from the High Performance Computing Center (HPCC) at Shanghai Jiao Tong University.

Author Contributions

C.Y. and Y.Z. conceived the research project. C.Y. and L.M. developed the program. C.Y., L.M., S.L., H.T. and Y.Z. analysed data and wrote the manuscript. Y.Z. and D.Y. supervised and supported the study. All the authors participated in discussions of the research and reviewed the manuscript.

Additional Information

Supplementary information accompanies this paper at <http://www.nature.com/srep>

Competing financial interests: The authors declare no competing financial interests.

How to cite this article: Yu, C. *et al.* *HBP Builder: A Tool to Generate Hyperbranched Polymers and Hyperbranched Multi-Arm Copolymers for Coarse-grained and Fully Atomistic Molecular Simulations.* *Sci. Rep.* **6**, 26264; doi: 10.1038/srep26264 (2016).



This work is licensed under a Creative Commons Attribution 4.0 International License. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in the credit line; if the material is not included under the Creative Commons license, users will need to obtain permission from the license holder to reproduce the material. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>