SUPPLEMENTARY INFORMATION
Hybrid Copper – Silver Nanoparticle Inks and Their Performance Under Thermal and Photonic Sintering

1.1 Energy-dispersive X-ray (EDX) Spectroscopy Data from Copper (Cu) – Silver (Ag) Nanoparticle (NP) Inks

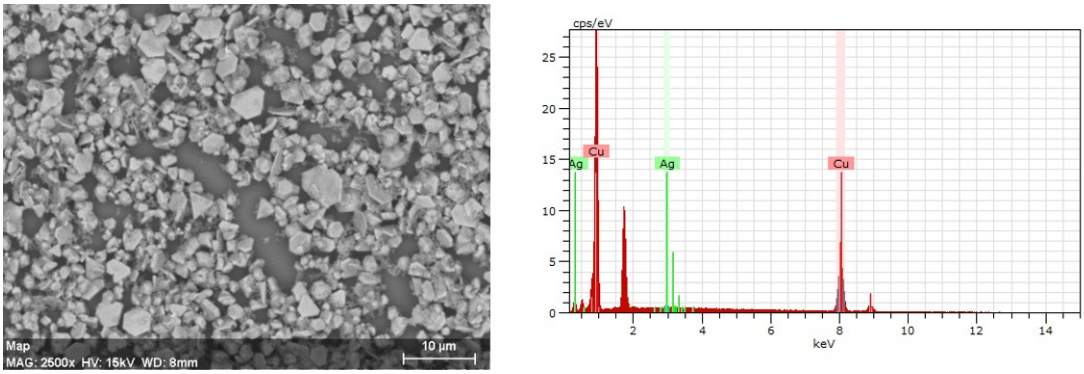

Figure S1. EDX data of dried 1/99 Ag/Cu NP ink by weight. Left: Backscatter SEM image of the analysed area. Right: EDX spectrum of the same area.

Element	Atomic Number	Series	Net	[Wt.%]	[Normalised Wt.%]	[Normalised At.%]	Error in %
COPPER	29	K-Series	22439	60.58	63.90	35.50	1.93
SILICON	14	K-Series	34466	21.66	22.84	28.72	0.92
CARBON	6	K-Series	2181	9.90	10.44	30.69	1.44
OXYGEN	8	K-Series	1479	2.10	2.22	4.89	0.34
SILVER	47	L-Series	704	0.57	0.60	0.20	0.04
			SUM:	94.81	100	100	

Table S1. Quantified EDX data from Figure S1.

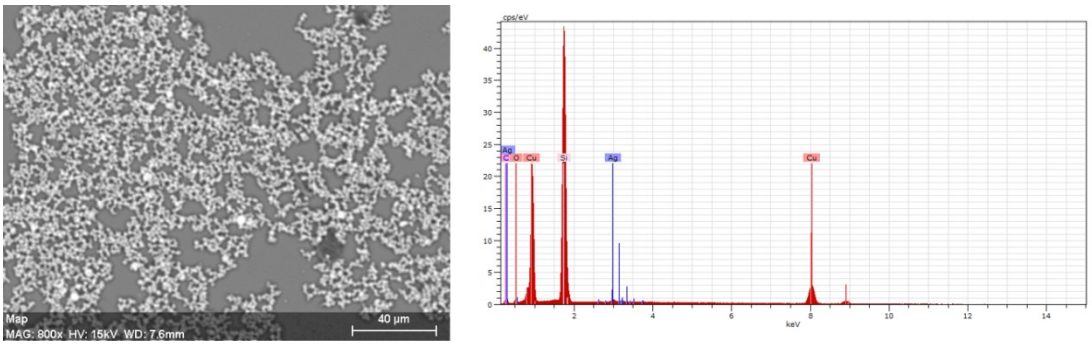

Figure S2. EDX data of 3/97 Ag/Cu NP ink by weight. Left: Backscatter SEM image of the analysed area. Right: EDX spectrum of the same area.

Element	Atomic Number	Series	Net	[Wt.%]	[Normalised Wt.%]	[Normalised At.%]	Error in %
SILICON	14	K-Series	144402	50.08	46.91	50.37	2.10
COPPER	29	K-Series	19233	40.97	38.38	16.42	1.18
CARBON	6	K-Series	2822	12.57	11.77	29.56	1.75
OXYGEN	8	K-Series	1423	1.87	1.75	3.29	0.31
SILVER	47	L-Series	1932	1.27	1.19	0.35	0.07
			SUM:	106.75	100	100	

Table S2. Quantified EDX data from Figure S2.

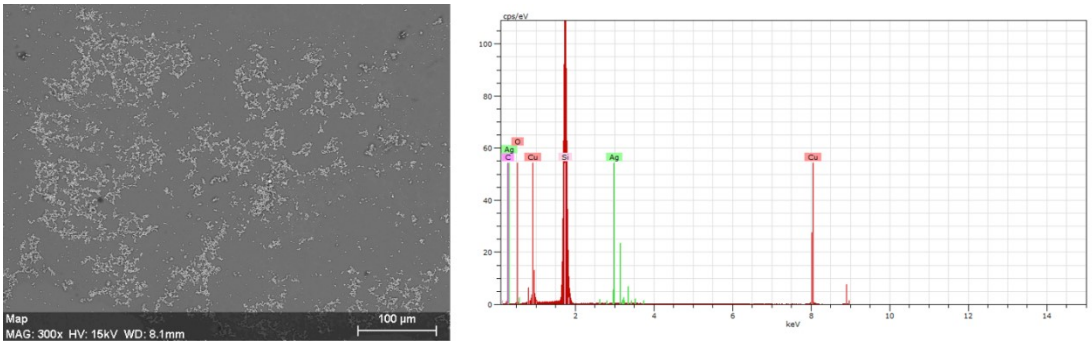

Figure S3. EDX data of 8/92 Ag/Cu NP ink by weight. Left: Backscatter SEM image of the analysed area. Right: EDX spectrum of the same area.

Element	Atomic Number	Series	Net	[Wt.%]	[Normalised Wt.%]	[Normalised At.%]	Error in %
SILICON	14	K-Series	199731	83.99	71.84	60.87	3.51
CARBON	6	K-Series	1763	19.21	16.43	32.56	2.86
COPPER	29	K-Series	2480	10.28	8.80	3.18	0.34
OXYGEN	8	K-Series	835	2.52	2.16	3.21	0.45
SILVER	47	L-Series	812	0.91	0.77	0.18	0.05
			SUM:	116.92	100	100	

Table S3. Quantified EDX data from Figure S3.

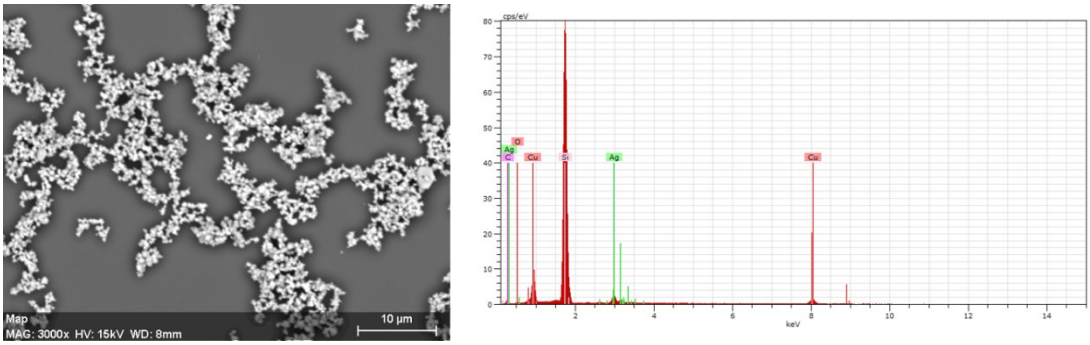

Figure S4. EDX data of 25/75 Ag/Cu NP ink by weight. Left: Backscatter SEM image of the analysed area. Right: EDX spectrum of the same area.

Element	Atomic Number	Series	Net	[Wt.%]	[Normalised Wt.%]	[Normalised At.%]	Error in %
SILICON	14	K-Series	118171	68.61	62.74	59.91	2.87
COPPER	29	K-Series	4429	18.66	17.06	6.72	0.57
CARBON	6	K-Series	1334	14.68	13.42	29.97	2.30
SILVER	47	L-Series	4999	6.22	5.69	1.58	0.24
OXYGEN	8	K-Series	330	1.19	1.09	1.82	0.27
			SUM:	109.36	100	100	

Table S4. Quantified EDX data from Figure S4.

Note that the presence of carbon and oxygen within the EDX analysis, as seen in Table S1-Table S4, originates from the substrate material.

1.2 Bar Coating Process

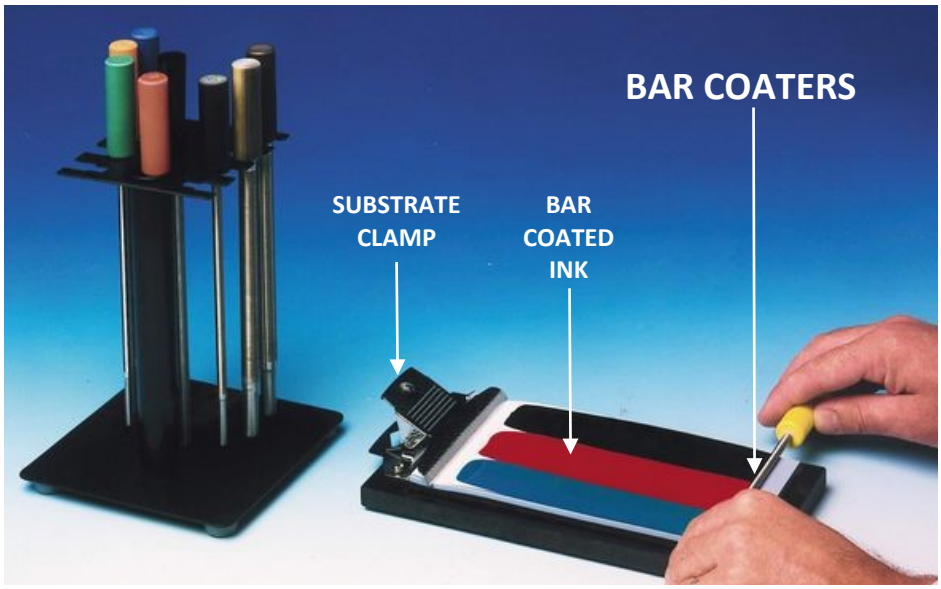

Figure S5: Photograph showing the bar coating of coloured inks onto paper substrates.[1]

1.3 Differential Scanning Calorimetry (DSC) Curve of Copper (Cu) NPs

DSC measurements were performed in order to determine the sintering temperature range for the copper nanoparticle samples. Particle sintering is an irreversible exothermic process during which particles give up surface energy in order to form bonds.[2] Particle sintering for the Cu-only material may therefore be identified through the exothermic peak (positive direction on the y-axis) in Figure S6 between approximately 155 and 597 °C. This peak only occurs on the first temperature ramp and does not occur during the second and third temperature ramps, as sintering is an irreversible one-way process. The broad particle sintering peak reflects the broad particle size distribution (see Figure 2b in the Main Text), with Cu NP diameters ranging from 101 to 1228 nm as measured *via* laser diffraction analysis. Since sintering temperature is dependent upon particle size - the smaller the particles, the lower the sintering temperature[2] – it follows that samples with a wide particle size distribution will also possess a relatively wide sintering temperature window.

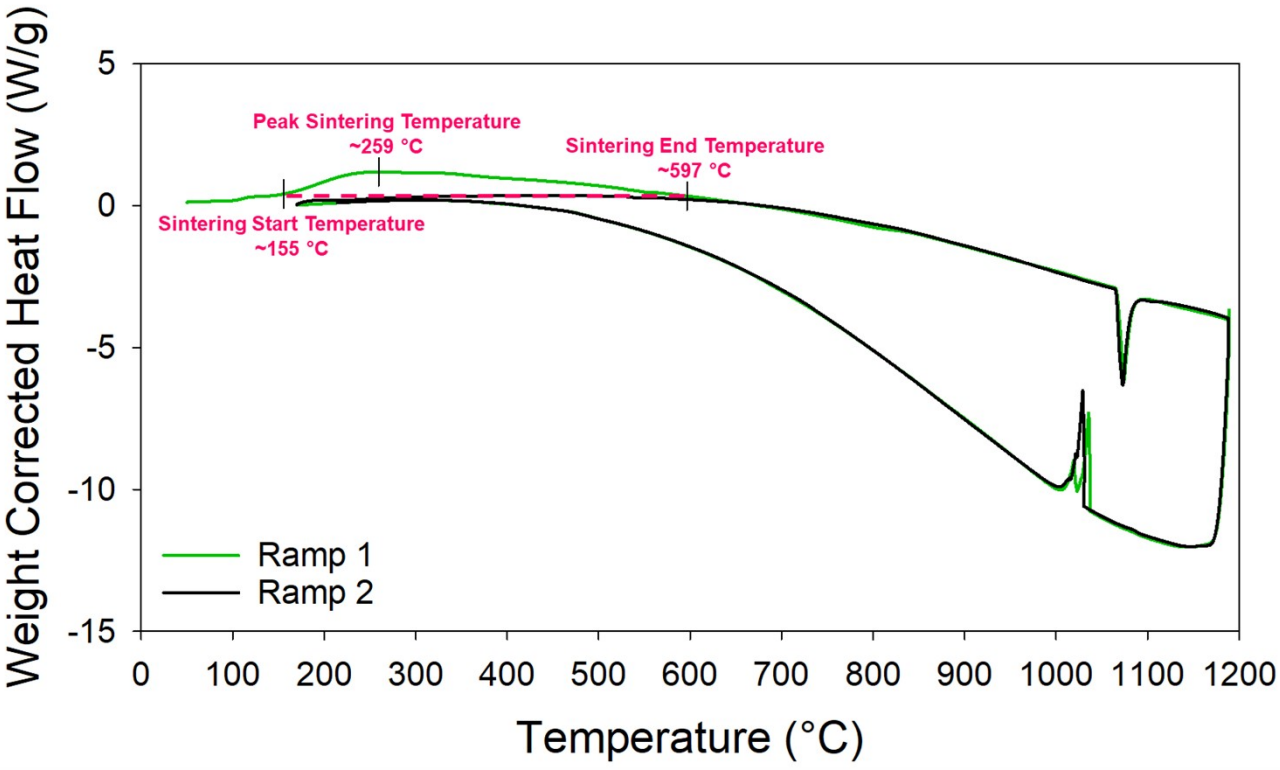

Figure S6. DSC curve of weight corrected heat flow against temperature for a Cu NP-only ink.

1.4 FEG-SEM Silver Nanoparticle Image Analysis

For the analysis of Ag NPs (see Figure 6 in the Main Text), SEM images were obtained using x35,000 magnification. Images were binarised using a global threshold. Continuous objects of fewer than 75 pixels were considered image noise and were removed. To obtain the number of particles in a frame, the MATLAB command "regionprops", which counts the number of continuous objects, was utilised. The number of particles in the frame prior to any heating (at room temperature) was averaged. Subsequent numbers of particles at increased temperatures were averaged and taken as a fraction of the average number of particles at room temperature. An example MATLAB script can be seen in the following Section 1.4.1.

1.4.1 Matlab Analysis of Sintering Silver (Ag) Nanoparticles

```
datedir = dir('C:\SEM Images');
filenames = {datedir.name};
n = numel(filenames);

for i = 3:n
    filename = datedir(i).name;
    im=imread(filename);
    cropped_im=imcrop(im,[1 1 1280 959]);

    %% binarise - global threshold
    bw = imbinarize(im_ cropped);

    bw_noiseremoved = bwareaopen(imbinarize,75); % remove all object
containing fewer than 75 pixels

    %% Determine particle properties
    stats = regionprops(bw_noiseremoved,'EquivDiameter');

    %% number of discrete particles
    no_p = 0;

    for j = 1:(numel(stats))
        if stats(j).Solidity > 0.9
            no_p = no_p+1;
        end
    end

    % save number of discrete particles
    sheet = 'Sheet1';
    xlRange = strcat('B',num2str(i-2));
    xlswrite('no_of_discrete_particles.xlsx',no_p,sheet,xlRange)
end
```

1.5 FEG-SEM Copper Nanoparticle Image Analysis

Cu NP samples were subjected to thermal treatment under vacuum using the host stage of an FEG-SEM (as explained in the Materials and Methods section of the Main Text). Secondary electron images were taken *in-situ* from the same sample location at regular temperature intervals to monitor NP sintering. Examples of a Cu NP cluster as well as an NP bed can be seen in Figure 5 of the Main Text and Figure S8, respectively. As the particle morphology and size distributions are significantly different for the Cu and Ag NPs, a different image analysis technique was employed to extract information from the Cu NP SEM images (see Section 1.4 above for details of the Ag image analysis).

For Cu sample analysis, the following manual highlighting methodology was chosen, as it enabled capture of overlapping particles. The SEM images were first printed out onto paper and an acetate was placed on top of the printout. A thin marker pen was used to highlight the edges of particles, as well as the scaling bar of the SEM image. The acetates with the highlighted particle edges were then scanned and analysed using MATLAB. The scanned images were converted to a black and white binary image using a global threshold. Continuous objects of fewer than 100 pixels were here considered image noise and were removed. Particle sizes were again obtained using the MATLAB command “regionprops”. A sphere with an equivalent area to the analysed object is used to calculate a particle diameter, which can then be scaled from number of pixels to the equivalent size in nanometres. Example image processing steps and MATLAB script are provided in the following Sections 1.5.1 and 1.5.2.

1.5.1 Image Analysis Processing Steps

Figure S7. Example of steps involved in particle size distribution quantification from left to right; Left: Secondary electron SEM image of Cu particles; Centre: Scan of the acetate with manually highlighted particles; Right: Image converted to binary for analysis.

1.5.2 Matlab Analysis of Sintering Copper (Cu) Nanoparticles

```
% Round objects analysis by David Pervan

%% Read image
filepath = ('400.1.tif');
image = imread(filepath);
figure(1); imshow(image);

%% Threshold image
bw = im2bw(image,0.55);
figure(3); imshow(bw);

%% Invert image
bw_inverted = ~bw;

%% Remove noise
% remove all object containing fewer than 100 pixels
bw_noiseremoved = bwareaopen(bw_inverted,100);

%% Determine particle size

stats = regionprops(bw_noiseremoved,'EquivDiameter');

folder = 'D:\SEM Images;
baseFileName = '400.1_stats.xlsx';
fullFileName = fullfile(folder, baseFileName);

writetable(struct2table(stats),fullFileName)

diameters = zeros(numel(stats),1); % CHANGE STUCTURE

for i = 1:(numel(stats))
    diameters(i) = stats(i).EquivDiameter; %ideal
end

diameters = sort(diameters);

scale_bar = (3438-3204)/1000; % apply scale bar from pixels to nanometres
diameters = diameters/scale_bar;

baseFileName = '400.1_diameters.xlsx';
fullFileName = fullfile(folder, baseFileName);
xlswrite(fullFileName,diameters)
```

For Cu particles involved in fusion during sintering (see following Section 1.6), SEM images from consecutive temperature steps were visually compared to identify the fusing and the fused particles. The edges of those particles were then manually highlighted using the software Image J Version 1.52a.

1.6 Investigation into the Thermal Sintering Process of Cu-Only Samples

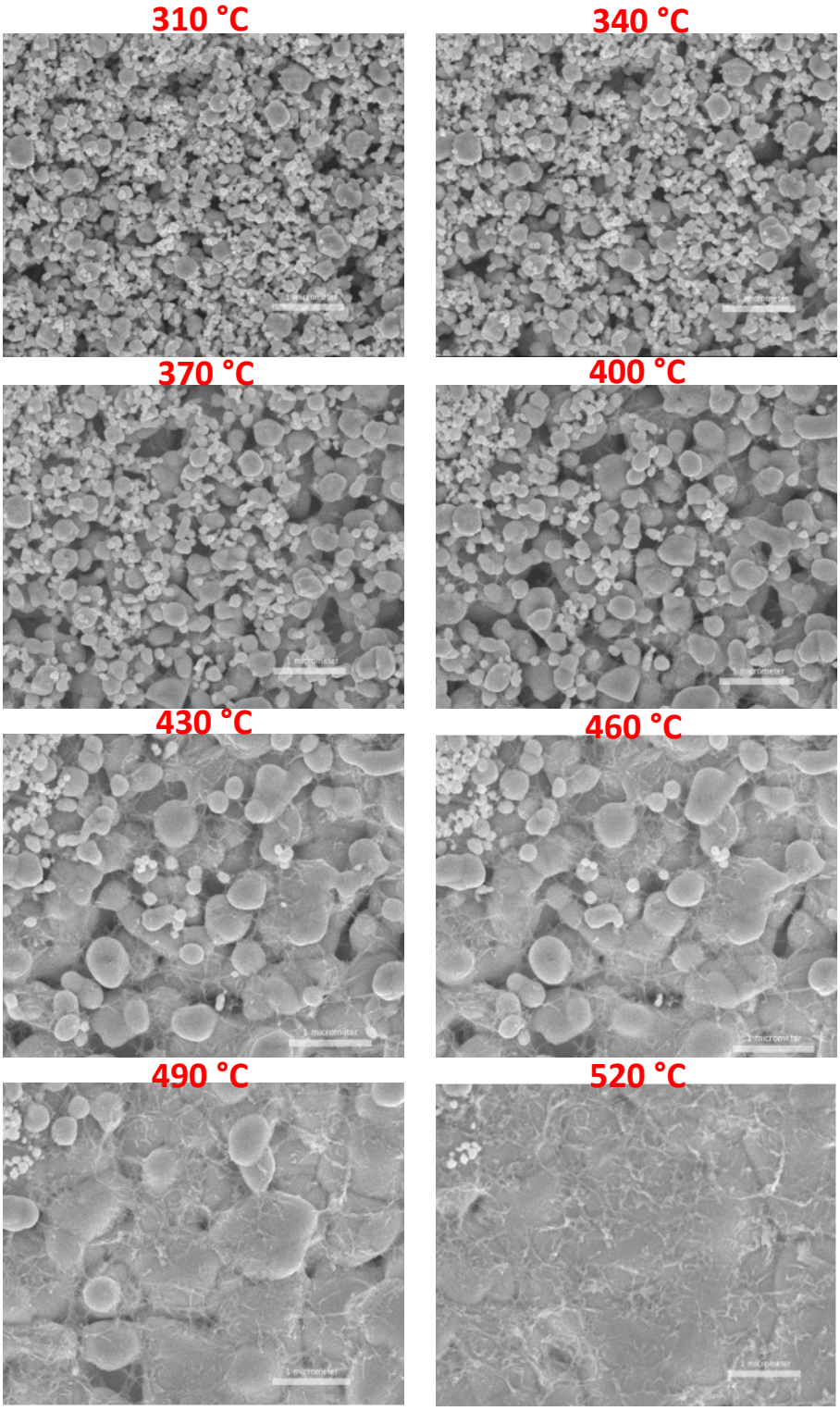

Figure S8. Secondary electron SEM images of a Cu NP bed, acquired from the same location at various temperature intervals, indicated by the red labels.

Figure 5 of the Main Text and Figure S8 show SEM images of a Cu NP cluster and Cu NP bed respectively, taken at regular temperature intervals during *in-situ* thermal sintering. Similar images from the *in*-situ sintering of Ag NPs are provided in Figure 6 of the Main Text and discussed therein.

Mittal et al.[3] have previously reported *in-situ* Cu NP sintering and simultaneous monitoring using a transmission electron microscope (TEM). However, in their case the analysis was entirely descriptive rather than quantitative, and rather difficult to follow due to the poor image quality. Quantitative analysis of Cu and Ag NP *in-situ* sintering processes made at regular temperature intervals is lacking within the literature.

The general limitation of taking top-down SEM images from one perspective is that information about the multi-layered morphology must be inferred. Error here can be minimised by increasing the particle count and thus increasing the probability of capturing a representative sample of the particle distribution.

Average Cu particle diameters were obtained at regular temperature intervals, from images such as those provided in Figure 5 of the Main Text and Figure S8, through the image analysis method discussed in the previous section, with representative data shown in Figure S9. It can be seen that with increasing temperature, particularly between 400 and 490 °C, both the average particle diameter and the standard deviation of the particle size distribution increase.

Inter-particle diffusion leads to coarsening when NPs are in contact during sintering.[4] Large particles will grow at the expense of small particles, consequentially increasing the average particle size distribution. The particles involved in this fusion process were distinguished during the Cu NP image analysis procedure previously described; a smaller particle fusing into a larger particle is labelled as a *fusing* particle and the larger receiving counterpart is described as a *fused* particle. Diameters of *fused* and *fusing* particles are displayed with changing temperature in Figure S10. Above 350 °C, the sizes of *fused* particles were unobtainable as it was impossible to distinguish from the 2D SEM images whether a *fused* particle was distinct and separate, or whether it had sintered with its neighbours.

With increasing temperature, the average size of *fused* particles does not change significantly. The average *fusing* particle size, as well as the standard deviation of the size distribution, both increase with increasing temperature. The latter finding is consistent with the thermodynamics of particle sintering, where the driving force for mass transport increases with decreasing particle size.[4] This is because the driving force for sintering is affected by the specific surface energy, which in turn increases with decreasing particle size.[5,6] Consequentially, to induce fusion between larger particles, a higher temperature is required in order to provide a sufficient driving force for mass transport.

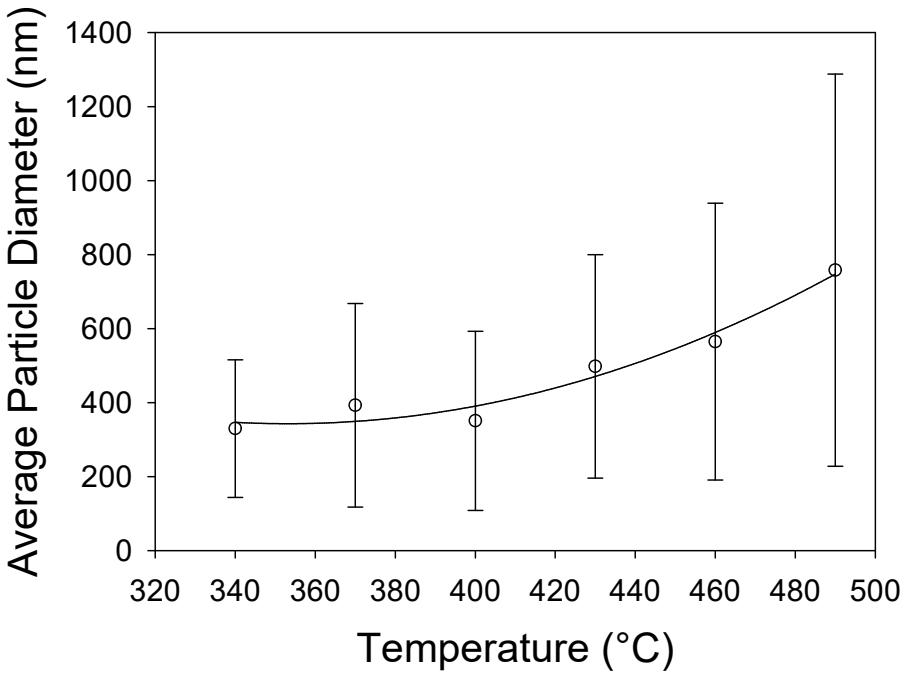

Figure S9: Average Cu particle diameter with increasing temperature. Error bars indicate standard deviation of the particle size distribution.

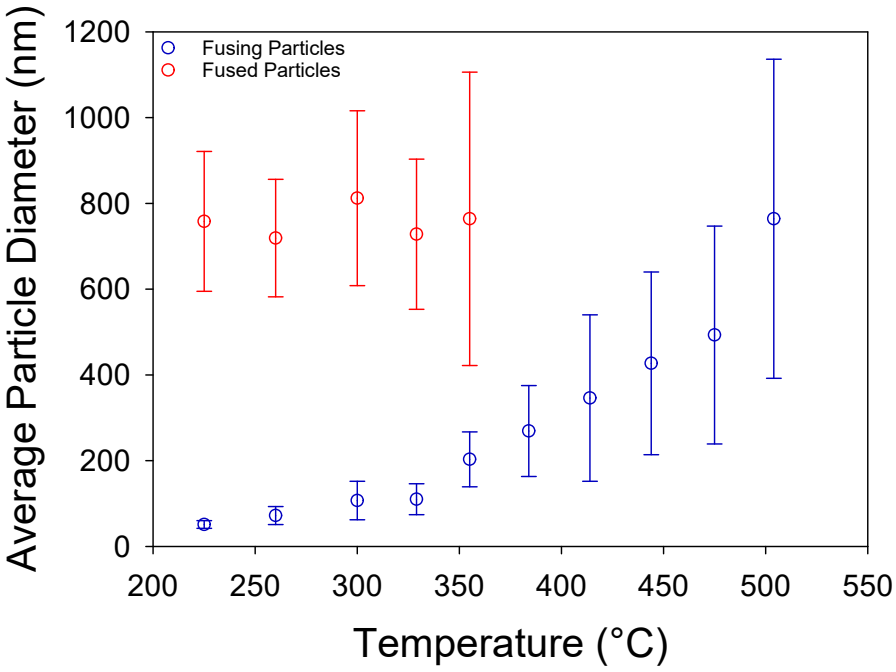

Figure S10: Average Cu particle size involved in particle fusion with increasing temperature. Error bars indicate standard deviation of the particle size distributions.

These observations are reflected in Herring's Scaling Law, which was first introduced in 1950.[7] The scaling law describes the effect of particle size on the sintering process, and takes into account the effects of temperature according to Equation 1.[8,9]

$$n\,ln\left(\frac{d_1}{d_2}\right) = \frac{Q}{R}\left(\frac{1}{T_2} - \frac{1}{T_1}\right) \qquad \text{Equation 1}$$

Where Q is the activation energy, R is the ideal gas constant, T_1 and T_2 are the corresponding sintering temperatures for particle diameters d_1 and d_2, and n is dependent upon the specific mass transport mechanism of the sintering phenomenon. For viscous flow, $n = 1$, $n = 2$ for evaporation and condensation mechanisms, $n = 3$ for volume diffusion, and $n = 4$ for grain boundary diffusion or surface diffusion.

It follows from Equation 1 that with an increasing ratio of adjacent particle sizes, the activation energy required to induce particle sintering increases. Comparing the SEM images taken from a bed of Ag NPS (Figure 6 in the Main Text) to the SEM images of Cu NPs in Figure S8, it can be seen that the particle size distribution appears narrower for the Ag particles i.e., there are larger variations in size between adjacent Cu particles when compared with Ag particles. This explains the lower sintering temperature range determined for the Ag NPs (170-200 °C, as discussed

in the Main Text), in contrast to the Cu NP sintering, which occurs mostly above 400 °C. Moreover, Ag NPs have an inherently lower sintering temperature as a result of the lower bulk melting point of silver (962 °C compared with 1085 °C for Cu). It is believed that surface diffusion is the main sintering mechanism responsible for the neck formation between Ag NPs.[10,11] During surface diffusion the sintered particles retain their initial structure with no reorientation. The driving force for sintering can be considered a chemical potential difference between facet surfaces and the neck region, leading to neck formation.[4] This is in contrast to the Cu NP sintering where, as discussed, smaller particles fuse entirely into adjacent larger particles.

1.7 Computational Simulation of Ag Accumulation in Adjacent Copper (Cu) Particle Necks

Theoretical simulations designed to computationally model the impact of Ag accumulation within the Cu NP necking region are provided in Figure 8 of the Main Text and discussed therein.

Those calculations were based on the following assumptions:

- The Cu particles are perfectly spherical.
- The outer shape of the silver neck is cylindrical.
- The electrical resistivities of the Cu and Ag are 1.68×10^{8} Ω m and 1.59×10^{8} Ω m, respectively.[12]

Various sintering states of two neighbouring Cu particles were considered, including differing degrees of prior Cu-Cu neck formation. As the Cu particle size distribution was found to be wide, the case of neighbouring Cu particles with varying diameters was also considered.

Within the simulations, electrical resistance was defined according to Equation 2.

$$R = \rho\frac{l}{A} \qquad \text{Equation 2}$$

Where R is the electrical resistance (Ω), ρ is electrical resistivity (Ω m), l is the length of the object (m), and A is the cross-sectional area of the object (m^2). In this case, the object of interest does not have a constant cross-sectional area (A) across the length of the object (l), *i.e.,* the cross-sectional area is a function of the position along the object (x) (see Main Text, Figure 8a).

In order to obtain an accurate representation of the dimensional term $\frac{l}{A}$ from Equation 2, an integral is taken of the function describing the cross-sectional area ($A(x)$) over the length (l) of the object. The following equation was used to calculate the inter-particle resistance.

$$R = \rho\int_{0}^{l}\frac{1}{A(x)}dx \qquad \text{Equation 3}$$

Where $A(x)$ describes the cross-sectional area as a function of the one-dimensional position along the object.[13,14]

The MATLAB scripts developed and utilised for the computational modelling are provided in the following Sections 1.7.1 and 1.7.2.

1.7.1 Adjacent Copper (Cu) Particles of Equal Size: MATLAB Code

```
clear all

d = 330e-9;
r = d/2;

Cu_sint_deg = asind(1/(2*4));
Cu_volume = ((4/6)*pi*(r.^3)) - ( (pi/3)*((r-(r*cosd(Cu_sint_deg)))^2)*(3*r-
(r-(r*cosd(Cu_sint_deg)))) );
Cu_contact_area = pi*(r*sind(Cu_sint_deg)).^2;

Cu_resistance = 1.68e-8;

Ag_resistance = 1.59e-8;

Ag_sint_deg = asind(1/(2*2));

no_of_points = 100;
no_of_steps = no_of_points-1;
interval = (Ag_sint_deg-Cu_sint_deg)/no_of_steps;

theta_for_plotting = zeros(no_of_points,1,1);

Ag_contact_area = zeros(no_of_points,1,1);
final_segment_Ag_faction = zeros(no_of_points,1,1);
segment_final_Ag_volume = zeros(no_of_points,1,1);
segment_final_volume = zeros(no_of_points,1,1);
segment_final_carea = zeros(no_of_points,1,1);
segment_final_length = zeros(no_of_points,1,1);

Ag_Cu_mass_ratio = zeros(no_of_points,1,1);
contact_area_increase = zeros(no_of_points,1,1);

segment_Cu_resistance = zeros(no_of_points,1,1);
final_segment_resistance = zeros(no_of_points,1,1);
resistance_overall = zeros(no_of_points,1,1);
segment_Cu_average_carea = zeros(no_of_points,1,1);
segment_Cu_length = zeros(no_of_points,1,1);

res_av = zeros(no_of_points,1,1);
res_improvement = zeros(no_of_points,1,1);

final_segment_resistance_new = zeros(no_of_points,1,1);
final_segment_resistance2 = zeros(no_of_points,1,1);
resistance_overall2 = zeros(no_of_points,1,1);

noAg_segment_Cu_length = r*cosd(Cu_sint_deg);

fun = @(L) L/((r*sind(acosd(L/r))).^2);
      xmin = 0;
      xmax = (r*cosd(Cu_sint_deg));
      int_allCu_noAg = integral(fun,xmin,xmax,'ArrayValued',true);

      noAg_segment_Cu_resistance = Cu_resistance * 2*int_allCu_noAg / pi;
int_Cu = zeros(no_of_points,1,1);
int_f_Cu = zeros(no_of_points,1,1);
```

```
int_f_Ag = zeros(no_of_points,1,1);

counter = 0;

for   theta = Cu_sint_deg:interval:Ag_sint_deg % Cu_sint_deg
      counter = counter+1;
      theta_for_plotting(counter) = theta;


      segment_Cu_length(counter) = r*cosd(theta);
      segment_final_length(counter) = (r*cosd(Cu_sint_deg)) -
(r*cosd(theta));
      segment_final_carea(counter) = pi*(r*sind(theta)).^2;
      segment_final_Cu_volume = (pi/6) * segment_final_length(counter) * (
3*(r*sind(theta)).^2 + 3*(r*sind(Cu_sint_deg)).^2 +
segment_final_length(counter).^2 );
      segment_final_volume(counter) = segment_final_length(counter) *
segment_final_carea(counter);
      segment_final_Ag_volume(counter) = segment_final_volume(counter) -
segment_final_Cu_volume;

      Ag_contact_area(counter) = segment_final_carea(counter)-
Cu_contact_area;
      contact_area_increase(counter) =
(segment_final_carea(counter)/Cu_contact_area)-1;

      Ag_Cu_mass_ratio(counter) =
(10490*segment_final_Ag_volume(counter))/(8960*Cu_volume);

      if theta ~= Cu_sint_deg % ensure it won't be "NaN"
            final_segment_Ag_faction(counter) =
      segment_final_Ag_volume(counter) / segment_final_volume(counter) ;
      else
            final_segment_Ag_faction(counter) = 0;
      end
      Cu_faction_in_final_segment = 1-final_segment_Ag_faction(counter);

      fun = @(L) L/((r*sind(acosd(L/r))).^2);
      xmin = 0;
      xmax = segment_Cu_length(counter);
      int_Cu(counter) = integral(fun,xmin,xmax,'ArrayValued',true);

      segment_Cu_resistance(counter) = Cu_resistance * 2*int_Cu(counter) /
pi;

      fun = @(L) L/((r*sind(acosd((r*cosd(Cu_sint_deg)-L)/r))).^2);
      xmin = segment_Cu_length(counter);
      xmax = (r*cosd(Cu_sint_deg));
      int_f_Cu(counter) = integral(fun,xmin,xmax,'ArrayValued',true);

      resistance_segment_final_Cu_part = Cu_resistance * 2*int_f_Cu(counter)
/ pi;

      fun = @(L) L/(((r*sind(acosd((r*cosd(Cu_sint_deg)-L)/r))).^2)-
((r*sind(Cu_sint_deg)).^2));
      xmin = segment_Cu_length(counter);
      xmax = (r*cosd(Cu_sint_deg));
```

```
      int_f_Ag(counter) = integral(fun,xmin,xmax,'ArrayValued',true);

      resistance_segment_final_Ag_part = Ag_resistance * 2*int_f_Ag(counter)
/ pi;

      final_segment_resistance(counter) = (
(1/resistance_segment_final_Cu_part) + (1/resistance_segment_final_Ag_part)
)^(-1);
     resistance_overall(counter) = segment_Cu_resistance(counter) +
final_segment_resistance(counter);

     res_improvement(counter) =
resistance_overall(counter)/noAg_segment_Cu_resistance;
end

figure(1)
hold on
plot(Ag_Cu_mass_ratio,res_improvement)
xlabel('Ag/Cu mass ratio')
ylabel('Resistance reduction due to Ag accumulation in Cu neck')
```

1.7.2 Adjacent Copper (Cu) Particles of Varying Size: MATLAB Code

```
clear all

l_d = 330e-9;
l_r = l_d/2;
particle_size_ratio = 1/1.0;
r_d = l_d*particle_size_ratio;
r_r = r_d/2;

l_Cu_sint_deg = asind(1/(2*8)); % must be smaller than r_r

if l_r*sind(l_Cu_sint_deg) >= r_r
      l_Cu_sint_deg = asind((r_r*.999)/l_r)
end

r_Cu_sint_deg = asind(l_r*sind(l_Cu_sint_deg)/r_r);

l_Cu_volume = ((4/6)*pi*(l_r.^3)) - ( (pi/3)*((l_r-
(l_r*cosd(l_Cu_sint_deg)))^2)*(3*l_r-(l_r-(l_r*cosd(l_Cu_sint_deg)))) );
r_Cu_volume = ((4/6)*pi*(r_r.^3)) - ( (pi/3)*((r_r-
(r_r*cosd(r_Cu_sint_deg)))^2)*(3*r_r-(r_r-(r_r*cosd(r_Cu_sint_deg)))) );
both_Cu_contact_area = pi*(l_r*sind(l_Cu_sint_deg)).^2;

Cu_resistance = 1.68e-8;
Ag_resistance = 1.59e-8;

l_Ag_sint_deg = asind(1/(2*3)); % must be smaller than r_r hence must be
smaller or equal to particle size ratio % must be bigger than l_Cu_sint_deg

if l_r*sind(l_Ag_sint_deg) > r_r
      l_Ag_sint_deg = asind(r_r/l_r)
end

r_Ag_sint_deg = asind(l_r*sind(l_Ag_sint_deg)/r_r);

no_of_points = 100;
no_of_steps = no_of_points-1;
interval = (l_Ag_sint_deg-l_Cu_sint_deg)/no_of_steps;

l_theta_for_plotting = zeros(no_of_points,1,1);
r_theta_for_plotting = zeros(no_of_points,1,1);
Ag_contact_area = zeros(no_of_points,1,1);
l_final_segment_Ag_faction = zeros(no_of_points,1,1);
r_final_segment_Ag_faction = zeros(no_of_points,1,1);

l_segment_final_Ag_volume = zeros(no_of_points,1,1);
l_segment_final_volume = zeros(no_of_points,1,1);
both_segment_final_carea = zeros(no_of_points,1,1);
l_segment_final_length = zeros(no_of_points,1,1);

r_segment_final_length = zeros(no_of_points,1,1);
r_segment_final_volume = zeros(no_of_points,1,1);
r_segment_final_Ag_volume = zeros(no_of_points,1,1);

Ag_Cu_mass_ratio = zeros(no_of_points,1,1);
```

```
contact_area_increase = zeros(no_of_points,1,1);

l_segment_Cu_resistance = zeros(no_of_points,1,1);
r_segment_Cu_resistance = zeros(no_of_points,1,1);
l_final_segment_resistance = zeros(no_of_points,1,1);
r_final_segment_resistance = zeros(no_of_points,1,1);

resistance_overall = zeros(no_of_points,1,1);
segment_Cu_average_carea = zeros(no_of_points,1,1);
l_segment_Cu_length = zeros(no_of_points,1,1);
r_segment_Cu_length = zeros(no_of_points,1,1);

res_av = zeros(no_of_points,1,1);
res_improvement = zeros(no_of_points,1,1);

final_segment_resistance_new = zeros(no_of_points,1,1);
final_segment_resistance2 = zeros(no_of_points,1,1);
resistance_overall2 = zeros(no_of_points,1,1);

fun = @(L) L/((l_r*sind(acosd(L/l_r))).^2);
      xmin = 0;
      xmax = (l_r*cosd(l_Cu_sint_deg));
      l_int_allCu_noAg = integral(fun,xmin,xmax,'ArrayValued',true);
      l_noAg_segment_Cu_resistance = Cu_resistance * l_int_allCu_noAg / pi;

fun = @(L) L/((r_r*sind(acosd(L/r_r))).^2);
      xmin = 0;
      xmax = (r_r*cosd(r_Cu_sint_deg));
      r_int_allCu_noAg = integral(fun,xmin,xmax,'ArrayValued',true);
      r_noAg_segment_Cu_resistance = Cu_resistance * r_int_allCu_noAg / pi;

overall_noAg_segment_Cu_resistance = l_noAg_segment_Cu_resistance +
r_noAg_segment_Cu_resistance;

l_int_Cu = zeros(no_of_points,1,1);
l_int_f_Cu = zeros(no_of_points,1,1);
l_int_f_Ag = zeros(no_of_points,1,1);

r_int_Cu = zeros(no_of_points,1,1);
r_int_f_Cu = zeros(no_of_points,1,1);
r_int_f_Ag = zeros(no_of_points,1,1);

counter = 0;

for   l_theta = l_Cu_sint_deg:interval:l_Ag_sint_deg % Cu_sint_deg
      r_theta = asind(l_r*sind(l_theta)/r_r);
      counter = counter+1;
      l_theta_for_plotting(counter) = l_theta;
      r_theta_for_plotting(counter) = r_theta;

      l_segment_Cu_length(counter) = l_r*cosd(l_theta);
      r_segment_Cu_length(counter) = r_r*cosd(r_theta);

      l_segment_final_length(counter) = (l_r*cosd(l_Cu_sint_deg)) -
(l_r*cosd(l_theta));
      both_segment_final_carea(counter) = pi*(l_r*sind(l_theta)).^2;
```

```
      l_segment_final_Cu_volume = (pi/6) * l_segment_final_length(counter) *
( 3*(l_r*sind(l_theta)).^2 + 3*(l_r*sind(l_Cu_sint_deg)).^2 +
l_segment_final_length(counter).^2 );
      l_segment_final_volume(counter) = l_segment_final_length(counter) *
both_segment_final_carea(counter);
      l_segment_final_Ag_volume(counter) = l_segment_final_volume(counter) -
l_segment_final_Cu_volume;

      r_segment_final_length(counter) = (r_r*cosd(r_Cu_sint_deg)) -
(r_r*cosd(r_theta));
      r_segment_final_Cu_volume = (pi/6) * r_segment_final_length(counter) *
( 3*(r_r*sind(r_theta)).^2 + 3*(r_r*sind(r_Cu_sint_deg)).^2 +
r_segment_final_length(counter).^2 );
      r_segment_final_volume(counter) = r_segment_final_length(counter) *
both_segment_final_carea(counter);
      r_segment_final_Ag_volume(counter) = r_segment_final_volume(counter) -
r_segment_final_Cu_volume;

      Ag_contact_area(counter) = both_segment_final_carea(counter)-
both_Cu_contact_area;
      contact_area_increase(counter) =
(both_segment_final_carea(counter)/both_Cu_contact_area)-1;

      Ag_Cu_mass_ratio(counter) =
(10490*(l_segment_final_Ag_volume(counter)+r_segment_final_Ag_volume(counter)
))/(8960*(l_Cu_volume+r_Cu_volume));

      if l_theta ~= l_Cu_sint_deg % ensure it won't be "NaN"
      l_final_segment_Ag_faction(counter) =
l_segment_final_Ag_volume(counter) / l_segment_final_volume(counter) ;

      else
            l_final_segment_Ag_faction(counter) = 0;
      end

      l_Cu_faction_in_final_segment = 1-l_final_segment_Ag_faction(counter);

      if r_theta ~= r_Cu_sint_deg % ensure it won't be "NaN"
            r_final_segment_Ag_faction(counter) =
      r_segment_final_Ag_volume(counter) / r_segment_final_volume(counter) ;


      else

            r_final_segment_Ag_faction(counter) = 0;

      end
      r_Cu_faction_in_final_segment = 1-r_final_segment_Ag_faction(counter);

      fun = @(L) L/((l_r*sind(acosd(L/l_r))).^2);
      xmin = 0;
      xmax = l_segment_Cu_length(counter);
      l_int_Cu(counter) = integral(fun,xmin,xmax,'ArrayValued',true);
      l_segment_Cu_resistance(counter) = Cu_resistance * l_int_Cu(counter) /
pi;

      fun = @(L) L/((r_r*sind(acosd(L/r_r))).^2);
      xmin = 0;
```

```
      xmax = r_segment_Cu_length(counter);
      r_int_Cu(counter) = integral(fun,xmin,xmax,'ArrayValued',true);
      r_segment_Cu_resistance(counter) = Cu_resistance * r_int_Cu(counter) /
pi;

      fun = @(L) L/((l_r*sind(acosd((l_r*cosd(l_Cu_sint_deg)-L)/l_r))).^2);
      xmin = l_segment_Cu_length(counter);
      xmax = (l_r*cosd(l_Cu_sint_deg));
      l_int_f_Cu(counter) = integral(fun,xmin,xmax,'ArrayValued',true);
      l_resistance_segment_final_Cu_part = Cu_resistance *
l_int_f_Cu(counter) / pi;

      fun = @(L) L/((r_r*sind(acosd((r_r*cosd(r_Cu_sint_deg)-L)/r_r))).^2);
      xmin = r_segment_Cu_length(counter);
      xmax = (r_r*cosd(r_Cu_sint_deg));
      r_int_f_Cu(counter) = integral(fun,xmin,xmax,'ArrayValued',true);
      r_resistance_segment_final_Cu_part = Cu_resistance *
r_int_f_Cu(counter) / pi;

      fun = @(L) L/(((l_r*sind(acosd((l_r*cosd(l_Cu_sint_deg)-L)/l_r))).^2)-
((l_r*sind(l_Cu_sint_deg)).^2));
      xmin = l_segment_Cu_length(counter);
      xmax = (l_r*cosd(l_Cu_sint_deg));
      l_int_f_Ag(counter) = integral(fun,xmin,xmax,'ArrayValued',true);
      l_resistance_segment_final_Ag_part = Ag_resistance *
l_int_f_Ag(counter) / pi;

      fun = @(L) L/(((r_r*sind(acosd((r_r*cosd(r_Cu_sint_deg)-L)/r_r))).^2)-
((r_r*sind(r_Cu_sint_deg)).^2));
      xmin = r_segment_Cu_length(counter);
      xmax = (r_r*cosd(r_Cu_sint_deg));
      r_int_f_Ag(counter) = integral(fun,xmin,xmax,'ArrayValued',true);
      r_resistance_segment_final_Ag_part = Ag_resistance *
r_int_f_Ag(counter) / pi;
     l_final_segment_resistance(counter) = (
(1/l_resistance_segment_final_Cu_part) +
(1/l_resistance_segment_final_Ag_part) )^(-1);


     r_final_segment_resistance(counter) = (
(1/r_resistance_segment_final_Cu_part) +
(1/r_resistance_segment_final_Ag_part) )^(-1);

     resistance_overall(counter) = l_segment_Cu_resistance(counter) +
l_final_segment_resistance(counter) + r_segment_Cu_resistance(counter) +
r_final_segment_resistance(counter);

     res_improvement(counter) =
resistance_overall(counter)/overall_noAg_segment_Cu_resistance;
end

figure(1)
plot(Ag_Cu_mass_ratio,res_improvement)
xlabel('Ag/Cu mass ratio')
ylabel('Resistance reduction due to Ag accumulation in Cu neck')
hold on
```

1.7.3 Additional Theoretical Simulations for the Neck Formation of Silver (Ag) and Copper (Cu) Nanoparticle (NP) Sintering

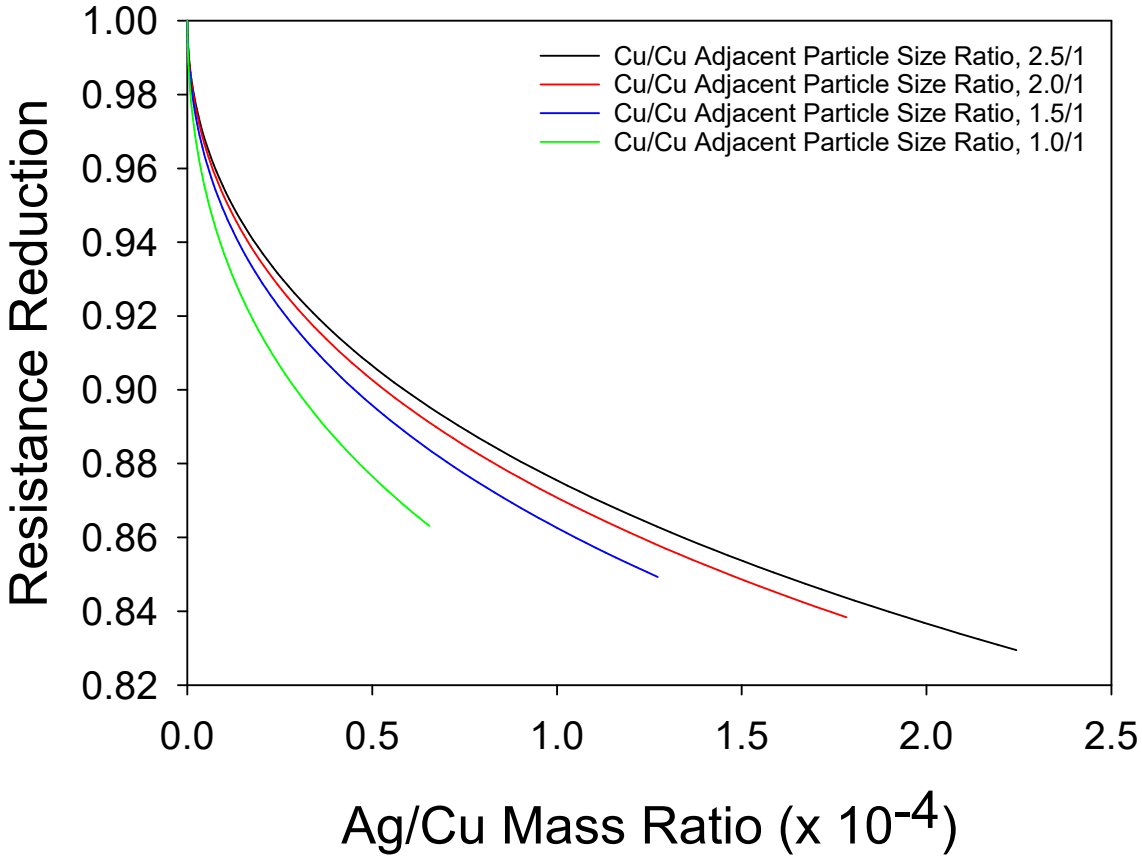

Figure S11: Theoretical resistance reduction between two Cu particles of differing sizes due to Ag accumulation in the Cu necking area with increasing Ag/Cu mass ratio for a Cu-Cu neck size of d/4, where d refers to the diameter of the larger Cu particle. Each curve ends when a maximum Ag neck filling of d/3 is reached.

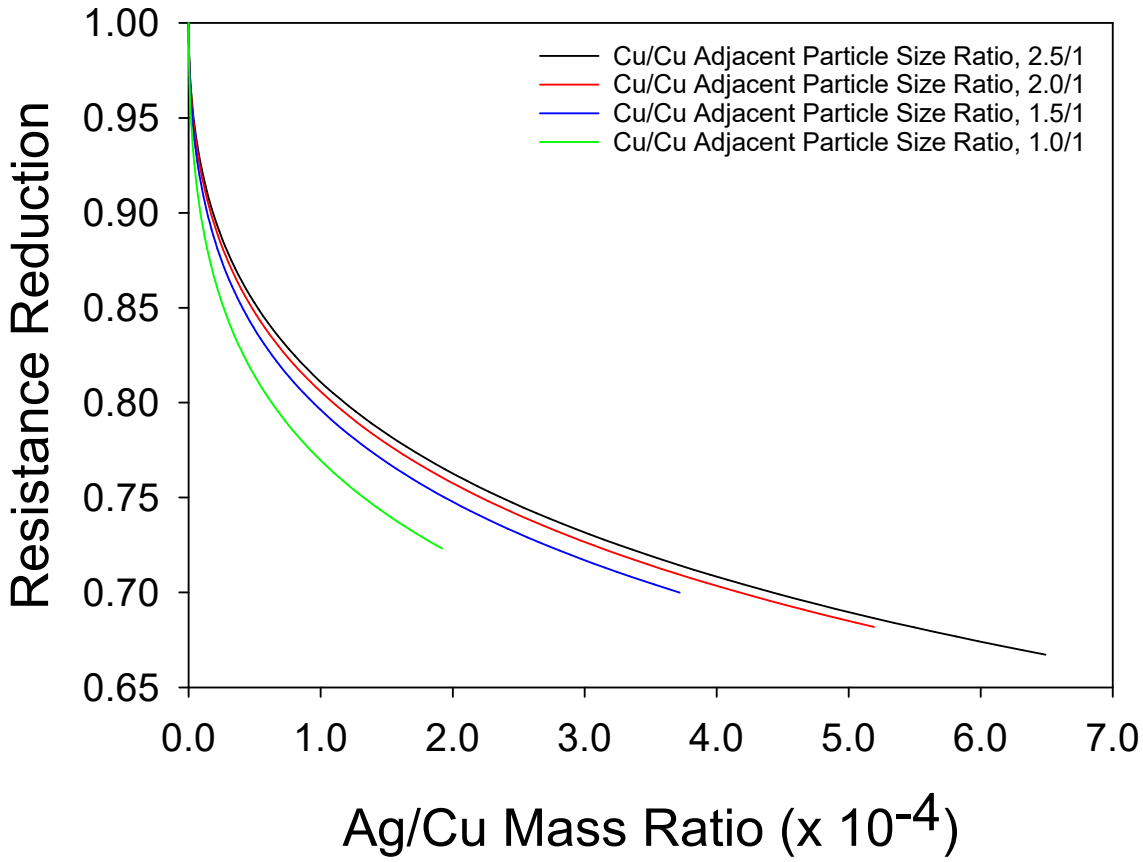

Figure S12: Theoretical resistance reduction between two Cu particles of differing sizes due to Ag accumulation in the Cu necking area with increasing Ag/Cu mass ratio for a Cu-Cu neck size of d/6, where d refers to the diameter of the larger Cu particle. Each curve ends when a maximum Ag neck filling of d/3 is reached.

References

1. RK Print. RK Print K Hand Coaters.
2. German, R. M. Thermodynamics of sintering. in *Sintering of Advanced Materials - Fundamentals and Processes* (ed. Fang, Z. Z.) 3–32 (Woodhead Publishing, 2010). doi:10.1533/9781845699949.1.3.
3. Mittal, J. & Lin, K.-L. Exothermic low temperature sintering of Cu nanoparticles. *Mater. Charact.* **109**, 19–24 (2015).
4. Fang, Z. Z. & Wang, H. Sintering of Ultrafine and Nanosized Particles. in *Sintering of Advanced Materials* (ed. Fang, Z. Z.) 434–473 (Woodhead Publishing, 2010).
5. Campbell, C. T., Parker, S. C. & Starr, D. E. The effect of size-dependent nanoparticle energetics on catalyst sintering. *Science (80-.).* **298**, 811–814 (2002).
6. Nanda, K. K., Maisels, A., Kruis, F. E., Fissan, H. & Stappert, S. Higher Surface Energy of Free Nanoparticles. *Phys. Rev. Lett.* **91**, 106102-1–4 (2003).
7. Herring, C. Effect of change of scale on sintering phenomena. *J. Appl. Phys.* **21**, 301–303 (1950).
8. Yan, M. F. & Rhodes, W. W. Low temperature sintering of TiO2. *Mater. Sci. Eng.* **61**, 59–66 (1983).
9. Messing, G. L. & Kumagai, M. Low-temperature sintering of α-alumina-seeded boehmite gels. (1994).
10. Bonevich, J. E. & Marks, L. D. Sintering behavior of ultrafine ceramic particles. *MRS Online Proc. Libr. Arch.* **286**, 3–8 (1992).
11. Shi, J. L., Deguchi, Y. & Sakabe, Y. Relation between grain growth, densification and surface diffusion in solid state sintering - A direct observation. *J. Mater. Sci.* **40**, 5711–5719 (2005).
12. Matula, R. A. Electrical resistivity of copper, gold, palladium, and silver. *J. Phys. Chem. Ref. Data* **8**, 1147–1298 (1979).
13. Meaden, G. T. *Electrical resistance of metals*. (Springer, 2013).
14. Rodríguez-Valverde, M. Á. & Tirado-Miranda, M. A simpler derivation of the integral formula of electrical resistance. *Eur. J. Phys.* **30**, L47 (2009).